

# Morrowind Scripting for Dummies

(9<sup>th</sup> Edition)

A manual for the TES Construction Set Scripting Language by  
**GhanBuriGhan**  
**Updated by Yacoby and melian**  
With help and information from the community



# Table of Contents

|                                                                |           |
|----------------------------------------------------------------|-----------|
| Foreword to the ninth edition .....                            | 9         |
| Foreword to eighth edition .....                               | 10        |
| <b>Introduction</b> .....                                      | <b>11</b> |
| <b>Using this guide</b> .....                                  | <b>11</b> |
| <b>What is a script?</b> .....                                 | <b>11</b> |
| <b>What can scripts do?</b> .....                              | <b>12</b> |
| <b>What scripts can't do:</b> .....                            | <b>12</b> |
| <b>Scripting tutorial</b> .....                                | <b>13</b> |
| <b>Let's get going!</b> .....                                  | <b>13</b> |
| <b>The scripting window</b> .....                              | <b>13</b> |
| <b>What do we want?</b> .....                                  | <b>14</b> |
| <b>Writing a script</b> .....                                  | <b>14</b> |
| Naming a script: Begin and End .....                           | 14        |
| Detecting an action by the player .....                        | 15        |
| Writing text and obtaining decisions from the player .....     | 15        |
| How local scripts are executed .....                           | 16        |
| Your first bug .....                                           | 19        |
| Putting a spell on the player .....                            | 19        |
| <b>How to learn more</b> .....                                 | <b>22</b> |
| <b>General Information: Scripts, Commands and Syntax</b> ..... | <b>23</b> |
| <b>Types of scripts</b> .....                                  | <b>23</b> |
| Local scripts .....                                            | 23        |
| Global scripts .....                                           | 23        |
| <b>Syntax</b> .....                                            | <b>25</b> |
| Beginning and ending scripts .....                             | 25        |
| General syntax for functions: .....                            | 25        |
| General syntax: Commas, parentheses and spaces .....           | 26        |
| Comments .....                                                 | 26        |
| Indentation / using tabstops .....                             | 26        |
| <b>Variables</b> .....                                         | <b>27</b> |
| Types of variables .....                                       | 27        |
| Local variables .....                                          | 28        |
| Global variables .....                                         | 29        |
| Referencing variables on other objects and scripts .....       | 29        |
| Using variables in functions .....                             | 30        |
| Operators / mathematical calculations .....                    | 30        |
| <b>Testing conditions</b> .....                                | <b>31</b> |
| Use of if... elseif conditions .....                           | 31        |
| While conditions .....                                         | 33        |
| Constructing Boolean operations in TES Script .....            | 34        |

|                                                                                   |           |
|-----------------------------------------------------------------------------------|-----------|
| <b>List of TES-script functions .....</b>                                         | <b>35</b> |
| Explanation of the format.....                                                    | 35        |
| <b>Working with objects.....</b>                                                  | <b>36</b> |
| <b>Working with inventory items .....</b>                                         | <b>36</b> |
| Adding and removing items from the inventory .....                                | 36        |
| Dropping items to the floor .....                                                 | 38        |
| Monitoring inventory activities: Adding, dropping items and using soul gems ..... | 39        |
| Force-equipping an Item .....                                                     | 40        |
| Detecting if an item has been equipped .....                                      | 41        |
| Disabling ability to equip an item .....                                          | 42        |
| Checking for presence of items in inventory .....                                 | 43        |
| Repairing objects.....                                                            | 43        |
| Worn / equipped object information .....                                          | 43        |
| UsedOnMe function .....                                                           | 46        |
| <b>Moving and placing objects.....</b>                                            | <b>46</b> |
| Moving along an objects axis.....                                                 | 46        |
| Moving along the world axis.....                                                  | 47        |
| CellUpdate.....                                                                   | 48        |
| Setting position (the other way to do movement) .....                             | 48        |
| Positioning an object in the world or in an interior cell .....                   | 50        |
| Resetting an object to its original position .....                                | 51        |
| Placing an item near the PC .....                                                 | 51        |
| Place items near an object .....                                                  | 52        |
| Creating new object-references with PlaceItem.....                                | 52        |
| <b>Rotation and angles.....</b>                                                   | <b>54</b> |
| Making an object spin .....                                                       | 54        |
| Setting Angles .....                                                              | 54        |
| <b>Scale Functions .....</b>                                                      | <b>56</b> |
| <b>Determining location, relative position and movement .....</b>                 | <b>58</b> |
| Detecting if player is indoors or outdoors.....                                   | 58        |
| Determining the players cell .....                                                | 58        |
| Distance of one object to another.....                                            | 59        |
| Determining an objects position and facing.....                                   | 59        |
| Line of Sight.....                                                                | 60        |
| <b>Determine whether an Actor is detected by another Actor.....</b>               | <b>62</b> |
| Determining when the PC leaves a cell.....                                        | 62        |
| Detect player traveling .....                                                     | 63        |
| Triggers for Actors standing on an object.....                                    | 64        |
| Hurting an Actor standing on an object .....                                      | 65        |
| Object Collision Functions.....                                                   | 65        |
| <b>Checking activation of an item and activating it.....</b>                      | <b>66</b> |
| <b>Locking and Unlocking doors or chests .....</b>                                | <b>69</b> |
| <b>Animating objects.....</b>                                                     | <b>69</b> |
| <b>Enabling and disabling objects .....</b>                                       | <b>71</b> |
| Deleting a reference completely.....                                              | 72        |
| <b>Don't save changes to an object .....</b>                                      | <b>74</b> |

|                                                                           |     |
|---------------------------------------------------------------------------|-----|
| <b>Scripting NPC's: AI and Movement</b> .....                             | 75  |
| <b>Make an NPC walk to a new location</b> .....                           | 75  |
| Checking whether an NPC has performed his movement.....                   | 75  |
| <b>Make an Actor turn or face a certain direction</b> .....               | 77  |
| <b>Define random Actor movement</b> .....                                 | 77  |
| <b>Making Actors activate objects</b> .....                               | 79  |
| <b>Following and Escorting</b> .....                                      | 80  |
| <b>Checking which AI package is currently executed</b> .....              | 81  |
| <b>Forcing sneak movement</b> .....                                       | 83  |
| <b>Forcing running and jumping: Tribunal NPC Movement Functions</b> ..... | 85  |
| <b>Detecting players action: running, jumping, sneaking?</b> .....        | 86  |
| <b>Detect combat readiness</b> .....                                      | 87  |
| <b>Making someone fall</b> .....                                          | 87  |
| <b>Equipment sharing and other companion functions</b> .....              | 88  |
| <b>Race, Faction and Rank</b> .....                                       | 90  |
| <b>Determining Race</b> .....                                             | 90  |
| <b>Determining the PC's Faction status:</b> .....                         | 90  |
| <b>Modifying faction standing and reaction</b> .....                      | 91  |
| Determining and changing reputation and disposition.....                  | 93  |
| <b>Werewolf-specific functions</b> .....                                  | 94  |
| Set the werewolf attributes .....                                         | 94  |
| Change the color of Secunda.....                                          | 94  |
| Determine how many kills a werewolf has .....                             | 94  |
| Check to see if the creature is in werewolf form.....                     | 95  |
| Change to a werewolf.....                                                 | 95  |
| Special werewolf global variables.....                                    | 96  |
| <b>Text and Dialogue</b> .....                                            | 97  |
| <b>Brief dialogue how-to</b> .....                                        | 97  |
| The concept of MW dialogue.....                                           | 97  |
| How dialogue works.....                                                   | 98  |
| A few golden rules .....                                                  | 101 |
| Dialogue 101 .....                                                        | 101 |
| <b>Dialogue-related functions</b> .....                                   | 103 |
| Displaying messages .....                                                 | 103 |
| Displaying variables and text defines in a message box .....              | 105 |
| Adding a dialogue topic .....                                             | 106 |
| Initiating and ending dialogue.....                                       | 107 |
| Allowing forced Dialogue with Werewolf player (Bloodmoon) .....           | 108 |
| Multiple choice – asking questions .....                                  | 109 |
| Adding to the journal and testing journal entries .....                   | 110 |
| Special dialogue-only functions.....                                      | 111 |
| Changing the Hello setting.....                                           | 112 |
| Useful dialogue variables .....                                           | 112 |

|                                                                              |                   |
|------------------------------------------------------------------------------|-------------------|
| <b><i>Changing and testing Skills, Attributes, and other Stats .....</i></b> | <b><i>113</i></b> |
| <b>Get, Set, and Modify stats - general remarks .....</b>                    | <b>113</b>        |
| <b>Determining and changing Actor and player stats: .....</b>                | <b>114</b>        |
| Determining and changing Attributes: .....                                   | 114               |
| Determining and changing Health, Magicka, Fatigue: .....                     | 114               |
| Determining and changing Skills: .....                                       | 115               |
| Determining and changing Level .....                                         | 115               |
| GetStat, ModStat and SetStat: A concerned modder's guide. - Galsiah .....    | 116               |
| <b><i>Combat.....</i></b>                                                    | <b><i>120</i></b> |
| <b>Initiating and ending combat .....</b>                                    | <b>120</b>        |
| <b>Detecting Attack .....</b>                                                | <b>120</b>        |
| <b>Combat related Get/Mod/Set AI functions: Fight, Flee, Alarm .....</b>     | <b>122</b>        |
| <b>Keeping track of kills and knockouts .....</b>                            | <b>123</b>        |
| <b>Resurrecting a dead Actor.....</b>                                        | <b>125</b>        |
| <b><i>Crime.....</i></b>                                                     | <b><i>126</i></b> |
| <b>Determining and changing Crime Level .....</b>                            | <b>126</b>        |
| <b>Jailing the PC.....</b>                                                   | <b>126</b>        |
| <b>Clearing the PC of crime .....</b>                                        | <b>126</b>        |
| <b>Detecting crime.....</b>                                                  | <b>126</b>        |
| <b>Useful global variables .....</b>                                         | <b>128</b>        |
| <b><i>Magic .....</i></b>                                                    | <b><i>129</i></b> |
| <b>Limiting the use of teleport .....</b>                                    | <b>129</b>        |
| <b>Limiting the use of levitation.....</b>                                   | <b>130</b>        |
| <b>Checking and managing souls and soulgems.....</b>                         | <b>131</b>        |
| <b>Adding and removing spells and cursing .....</b>                          | <b>133</b>        |
| <b>Casting spells .....</b>                                                  | <b>134</b>        |
| Managing and testing for spells .....                                        | 135               |
| Managing and testing spell effects .....                                     | 136               |
| Testing disease .....                                                        | 137               |
| <b>Explosion .....</b>                                                       | <b>138</b>        |
| <b>Magic Get/Mod/Set effects functions: .....</b>                            | <b>138</b>        |
| <b><i>Sound .....</i></b>                                                    | <b><i>140</i></b> |
| <b>Make Actors speak an audio file.....</b>                                  | <b>140</b>        |
| <b>Playing music .....</b>                                                   | <b>140</b>        |
| <b>Playing sounds .....</b>                                                  | <b>141</b>        |
| Controlling sound.....                                                       | 141               |
| Sound file formats .....                                                     | 142               |
| <b><i>Keeping track of time .....</i></b>                                    | <b><i>143</i></b> |
| <b>Timer .....</b>                                                           | <b>143</b>        |

|                                                          |            |
|----------------------------------------------------------|------------|
| <b>Morrowind's time related global variables.....</b>    | <b>143</b> |
| Keeping track of days passed .....                       | 144        |
| Moon phases.....                                         | 144        |
| <b>Weather.....</b>                                      | <b>146</b> |
| Changing weather .....                                   | 146        |
| Changing weather settings for a region.....              | 146        |
| Determining current weather.....                         | 147        |
| Detecting wind speed.....                                | 147        |
| Player sleeping.....                                     | 148        |
| Enabling and disabling player control and interface..... | 149        |
| Disable player control functions .....                   | 149        |
| Enable player control functions.....                     | 149        |
| Check player control status .....                        | 150        |
| Force first or third person view .....                   | 150        |
| Functions for character generation menus .....           | 150        |
| Determining if player has menus open.....                | 151        |
| Using MenuTest to open and close menus .....             | 152        |
| <b>Miscellaneous functions and variables .....</b>       | <b>153</b> |
| Breaking of script processing.....                       | 153        |
| Controlling global scripts .....                         | 153        |
| Fading the screen in and out .....                       | 154        |
| Adding a location to the map .....                       | 154        |
| Assigning random values to variables .....               | 154        |
| Playing videos .....                                     | 155        |
| Levelled List functions.....                             | 156        |
| Square root.....                                         | 157        |
| Water Level Functions.....                               | 157        |
| <b>Tips and tricks .....</b>                             | <b>160</b> |
| Little helpers: Text search, copy, paste .....           | 160        |
| Alternative scripting editors.....                       | 160        |
| Script Extenders .....                                   | 161        |
| Script with style for safer scripting .....              | 162        |
| Cleaning up your mod.....                                | 163        |
| On References Persist .....                              | 165        |
| The "72-Hours Bug": A Brief Explanation .....            | 165        |
| Limits of the Script Editor.....                         | 166        |
| Pitfalls.....                                            | 167        |
| Saving CPU time.....                                     | 168        |

|                                                                            |                   |
|----------------------------------------------------------------------------|-------------------|
| <b>Targeted scripts: running "global" scripts tied to an object .....</b>  | <b>170</b>        |
| <b>Detecting when the player does a load from saved game: .....</b>        | <b>171</b>        |
| <b>Uses of the CharGenState variable - Disabling saving and menus.....</b> | <b>173</b>        |
| <b>Detecting use of scrolls or books .....</b>                             | <b>174</b>        |
| <b>Making Actors switch between weapons.....</b>                           | <b>176</b>        |
| <b>Making NPCs switch between spell "sets" .....</b>                       | <b>177</b>        |
| <b>Making Actors lie down.....</b>                                         | <b>177</b>        |
| <b>Scripted teleporting.....</b>                                           | <b>181</b>        |
| <b>Interaction between mods.....</b>                                       | <b>183</b>        |
| <b>Safely starting global scripts- avoiding the main script.....</b>       | <b>184</b>        |
| <b>Use sound to detect events .....</b>                                    | <b>185</b>        |
| <b>Large battles .....</b>                                                 | <b>185</b>        |
| <b>A guide to making rideable objects .....</b>                            | <b>186</b>        |
| Selecting objects.....                                                     | 186               |
| Creating/Deleting objects .....                                            | 186               |
| Falling off from objects.....                                              | 187               |
| Collision detection.....                                                   | 188               |
| Savegame issue .....                                                       | 188               |
| <b>Trigonometry script - fast sine and cosine .....</b>                    | <b>189</b>        |
| <b>Mannequins.....</b>                                                     | <b>198</b>        |
| <b>Is she looking at me? .....</b>                                         | <b>201</b>        |
| <b>Cinematic sequence .....</b>                                            | <b>203</b>        |
| <b><i>Troubleshooting.....</i></b>                                         | <b><i>204</i></b> |
| <b>General hints.....</b>                                                  | <b>204</b>        |
| <b>The Console.....</b>                                                    | <b>204</b>        |
| Using the Console to check variables: .....                                | 204               |
| Using the Console to quickly test scripts: .....                           | 204               |
| <b>Error messages, malfunctions and common causes .....</b>                | <b>205</b>        |
| In the editor .....                                                        | 205               |
| In game error messages:.....                                               | 205               |
| <b><i>Appendix .....</i></b>                                               | <b><i>208</i></b> |
| <b>New functions that come with TRIBUNAL .....</b>                         | <b>208</b>        |
| Changes / fixes to Morrowind scripting:.....                               | 208               |
| Index of new Tribunal Script Functions:.....                               | 208               |
| <b>New functions that come with BLOODMOON .....</b>                        | <b>209</b>        |
| Index of new Bloodmoon Script Functions and variables: .....               | 209               |
| <b>Previously undocumented functions.....</b>                              | <b>210</b>        |
| <b>Variable-type functions: .....</b>                                      | <b>211</b>        |
| Local variables that get set by the game: .....                            | 211               |
| Local variables that you can set as a flag:.....                           | 211               |

|                                             |                   |
|---------------------------------------------|-------------------|
| Special Globals.....                        | 211               |
| <b>Game units: .....</b>                    | <b>213</b>        |
| <b>Derived attribute calculations:.....</b> | <b>213</b>        |
| <b>Magic Effect List .....</b>              | <b>215</b>        |
| <b>List of console commands .....</b>       | <b>216</b>        |
| <b>Game Settings .....</b>                  | <b>219</b>        |
| <i><b>Index .....</b></i>                   | <i><b>226</b></i> |

## Foreword to the ninth edition

Another update, this time by more than one person, now you can get really confused when it mentions that “I” did something or found something \*grins\*. Although this update was intended as a bugfix, a huge amount of new information has been added - since much of it concerns the bugs and pitfalls of the scripting language that have been discovered since version 8, you will find changes and additions scattered throughout.

I have also added a couple more functions that were missing from the last version, and there is a small section on script extenders to give a brief overview of what can be done with each which should hopefully prove informative to anyone wanting to know more about what they do. Galsiah provided a large section on the failings of get/mod/set stat; there is a lot more info on dialog and how it works, which should be very useful; a short explanation of float values and how they work has been added; some sections have been rearranged for clarity; and I *think* I've removed all the en dashes and typographer's quotes from code blocks, so it should be safe to copy-paste most of the example scripts now.

Special thanks go to GhanBuriGhan, who gave me permission to update his indispensable guide. Also thanks to Dave Humphrey for his permission to use the information on UESP wiki, and everyone who contributed to that wiki. I would like to thank exclusiveor77, DinkumThinkum, Björn, abot, Galsiah, Casey Tucker, cyran0 and JOG for the huge amount of information they gave me on the forums.

However without the help of the rest of community, everyone who collected all the information, posted their experiences, this guide would have never happened.

Yacoby and melian

## Foreword to eighth edition

Oh no, MSFD updates again! Quite a few changes this time. I have continued to reorder functions to achieve a more consistent classification. In the course of this restructuring of the text, Tribunal and Bloodmoon functions are now also sorted among the rest of the functions (but clearly marked), and the same applies for Get/Set/ModStat type functions - the ones that affect magic are now in the Magic section, the ones that affect combat are in the combat chapter, and so on. I reformatted all the function entries for better visibility. There are also a good number of corrections and additions: check out the corrected list of idles in AIWander, more detailed info on functions such as OnDeath, OnActivate, or the revelation that some functions, like PositionCell use a different unit for the angle setting (minutes instead of degrees). There is a lot of interesting new detail on the intricacies of RemoveItem, OnPCEquip, and SkipEquip. Throughout, I have added a couple more sample scripts. I have assembled a tips list on Dialogue from the information from a once-pinned thread on the official forums. To make your mod-testing easier, I now have a much more complete list of console commands, and you may find that some of these may actually be useful script functions as well. A nice collection of new tips and tricks have also been added, e.g. on how to detect savegame loading, alternative script editors, a fast sine and cosine calculator and more.

I would like to mention something here, that has become more and more evident in recent discussions. Apparently the different versions of the game (Morrowind, the expansions, various updates, European/American versions), have quite a number of functional differences with respect to scripting. This may explain a lot of the conflicting information I get about some functions. Some functions actually seem to have lost functionality in later versions of the game, some regained functionality, etc. In the absence of a full change log by Bethesda it is unfortunately impossible to really give a final verdict in many of these cases.

Again I can only stress that this guide would never have become what it is without the help and support of the modding community. This time, I would like to say special thanks to Nigedo, DinkumThinkum, ManaUser, ThePal, Erstam, JDGBOLT, Klinn, and MentalElf, for their tireless help in finding and solving many errors or inconsistencies in the last edition, or sharing scripts and information with me. Thanks also to Emma, for all the info she put into that dialogue thread on the forums.

GhanBuriGhan

*Disclaimer & Copyright Statment: The Elder Scrolls, Morrowind, Daggerfall, Arena, Tribunal, Bloodmoon the TES Construction Set etc. are property of Bethesda Softworks, a ZeniMax Media company. The authors and editors of this manual make no claims to these names and trademarks. In all other respects I (GhanBuriGhan) maintain the copyright for this document. This guide is fully unofficial and in no way am I, any other authors or editors, or Bethesda responsible for any damages or loss of data, patience, hair, or sanity inflicted through the use of this manual. You can freely distribute this document as long as you keep it intact and unmodified.*

**If you find older versions of this document on the web for download, please inform the webmaster about the new version.**

# Introduction

## ***Using this guide.***

If you are new to scripting, you should probably start by reading this introduction and especially do the tutorial. It is my best attempt at explaining scripts, what they do, and how to program one, in simple terms.

Secondly this is a manual, a reference, a handbook. The bulk of this document is a documentation of the available functions. This part is not written for the beginner, you are expected to know the basics of scripting already. I have however tried to provide a lot more info, explanations and sample scripts than the original helpfile, to make it easier to use these functions correctly in your scripts. Use the index at the end of the document to find info on a specific function, or the table of contents to find functions related to a general area of interest. If you run into errors, the Troubleshooting section may help you a bit. In the Appendix you find lists that may also come in handy as a reference.

Thirdly, as an advanced scripter you may find the Tips and Tricks section interesting that covers both basic advice and advanced scripting techniques.

Finally a word of advice: don't take what's written here as gospel. The info in this guide represents the best of my knowledge and forum wisdom, but that doesn't mean that there aren't mistakes and oversights, etc. E.g. if I write that a function doesn't take variables as arguments it *probably* doesn't – but if it is important for your mod, by all means try anyway. It may have changed with a patch or maybe simply nobody checked before. So run a test, and if you find something new, let me know, and I will add the info in a possible future update.

## ***What is a script?***

Scripts are basically pieces of code written in a special scripting language (I will call it TES script from here on). These little "programs" will run during the game and can perform certain things in the game, lots of things actually: Trigger events, control time and place, make things and creatures vanish, appear or move, give messages to the player, change stats, even change the weather – the possibilities are great.

TES Script is a unique scripting language, it is not used outside the TES Construction Set. As a scripting language it has certain limitations compared to a "real" programming language, like, e.g. C++:

1. The scope of TES Script is limited– don't expect to be able to program anything that is not already in the game in one way or another – which is not to say you can not achieve new and unusual things with scripting! But you can't use TES script to, say, program a word processor.
2. TES Script is also not an SDK (software development kit) that lets you actually work with and change (parts of) the games source code. That's why you can't use TES script to, for example, add new weather-effects. Those are hardcoded and you would need to change the actual game .exe to do that.
3. It's an interpreted language not a compiled one – the code needs a separate program (in this case Morrowind) to run – unlike compiled code that could be run by itself, like an .exe application.

### ***What can scripts do?***

Scripts for Morrowind are a way to have the game dynamically react to what the player does in the game world. You can use scripts to manage complex quests. You can use scripts to create custom items that perform actions beyond what regular enchantments could. You can use scripts to create traps. You can use scripts to change NPC or creature behavior.

Remember the character creation in Morrowind? It's basically controlled by a number of scripts. Seen Fargoth sneak around to his stash in Seyda Neen? That's a script directing his moves. Freed any slaves? That also is handled by a script. So the short answer to the question is: a lot.

### ***What scripts can't do:***

The TES script language is limited in its capabilities – there are only so many functions that you can use and sometimes the possible uses are not all that you may wish them to be. In fact some functions are buggy or plain broken. In many cases smart scripters can find workarounds for apparent limitations, but don't expect miracles. Many things are hardcoded and can not or only indirectly be influenced by scripts.

## Scripting tutorial

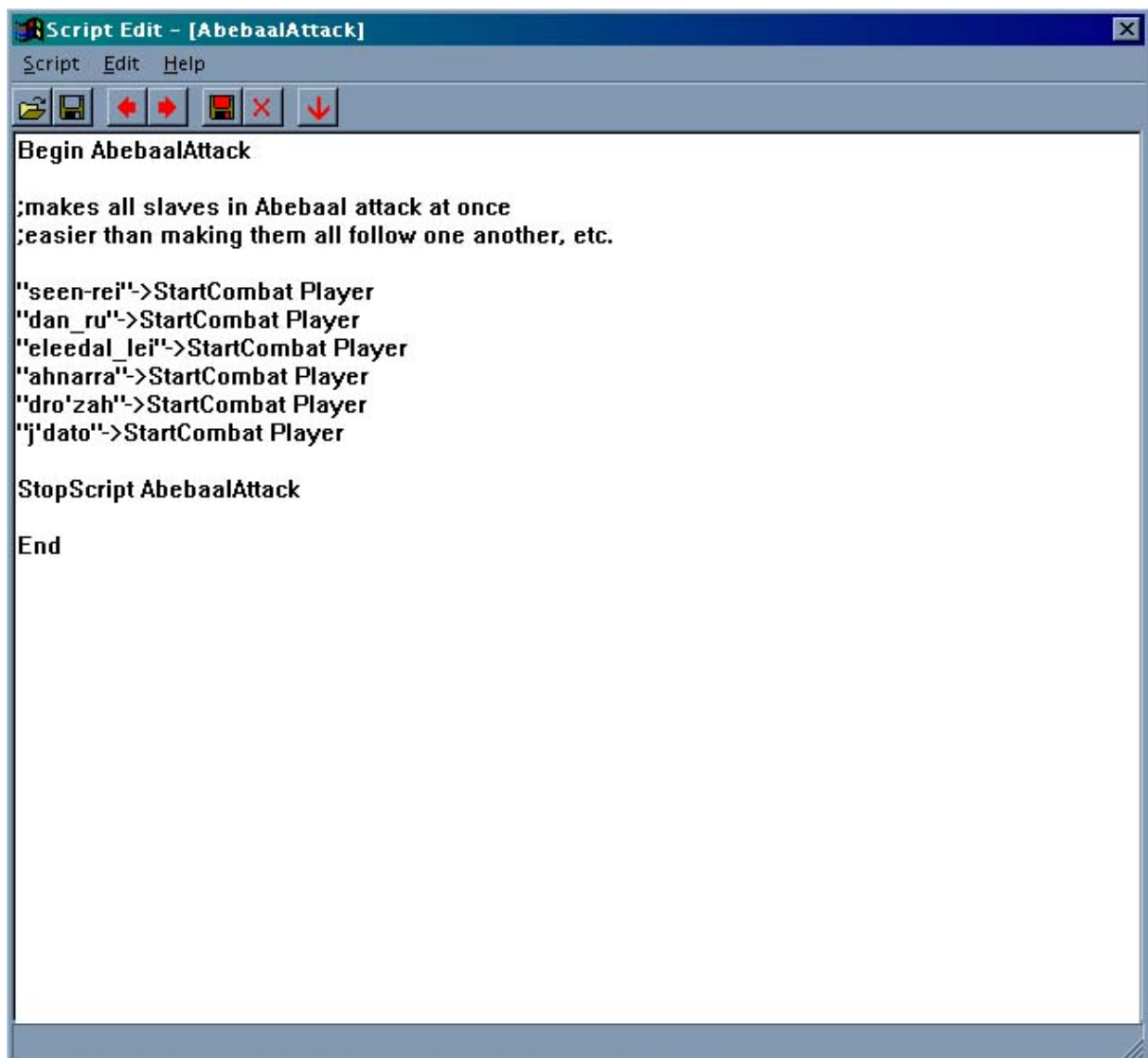
If you are completely new to scripting and programming in general, the thought of using TES script might be a little daunting – I have therefore written an extended tutorial that will walk you through making your first script. I will also explain the main elements of the scripting language as we go. There will be other explanations on the way, but the key instructions will be in **bold print**.

### *Let's get going!*

We start by opening the script editor: **Start up the TES Construction Set, open the Morrowind.esm file and then select Edit Scripts from the Gameplay menu to open the scripting window.**

### *The scripting window*

You enter the script editor either by **selecting Gameplay – Edit Scripts from the menu**, by clicking the edit script button (the pencil) in the taskbar or by accessing it from an Object or NPC dialogue, by clicking the button with the ellipses [...] next to the script field. The editor window is pretty basic:



Let's have a look at the buttons in the taskbar, from left to right: *Open* lets you select a script to edit. *Save* error checks the current script and compiles it or gives out error messages – note, however, that the plugin and thus the script is not really saved to disk at this time. When programming large scripts frequently use the save command in the main TESCS window after you have saved the script here, just in case the TESCS crashes.

Note that if you edit the script and suddenly hit "save plugin" to backup in the middle of the work, your updated script will NOT be saved with it. You must save it manually first. Also, if you just close script window, it doesn't mean that script will be saved. You must take care of it yourself (Thanks to Kir for this tip).

*Forward and Backward* arrows jump to the next or previous script, respectively (alphabetical order). If you give your scripts a common tag, that will make it easier to jump between the different scripts of your project, e.g. start every script name with *AA\_Scriptname* this will put them right at the beginning of the list and keep them neatly together. *Compile all* recompiles all scripts (what's this good for? I don't really know). Finally, the *delete* button deletes a script and the last "*arrow down*" button closes the script window.

The help menu gives quick access to the function and command pages of the helpfile (of moderate utility, hence the creation of this manual!)

You can cut, copy, and paste from and into the editor window by using the Windows standards ctrl-c for copy, ctrl-x for cut and ctrl-v for paste.

## ***What do we want?***

Before we really start writing our tutorial script we should decide what we want it to do. For this tutorial I decided we are going to make a **Riddle Chest: The chest will ask a riddle and only the right answer will open the chest. If the player provides the wrong answer, a trap will go off, hurting the player, and the chest can't be opened.**

That's a fairly complex undertaking, but we will take it step by step.

## ***Writing a script***

Ok, once you've got the Edit Script window open, **click into the main part of the window.** That is where you will write your script.

## **Naming a script: Begin and End**

First of all we must give our script a name – every script must start with the declaration of this name. **So please type:**

```
Begin my_first_script
```

**into the editor window.** Note the underscores: Your name should be one word. Also note that the script language is not case sensitive, so *Begin* could also be written without the capital letter: *begin*. This name is the handle by which the script will be known in the TESCS. Try hitting the save button now: you will get an error message about "you need to end your script with end scriptname."

So, for the editor to recognize the script we also need to indicate an end: **next, write**

```
End
```

**in a new line below the above.** As you see we can omit putting the name of the script in this line again, just *end* will do. When you hit *save* now, you will see the name appear in the title bar of the script editor, indicating that the script has been accepted. This is the shortest script possible - and of course it doesn't do anything at all.

## Detecting an action by the player

Next, we need a way to determine whether the player tries to open the chest. In TES Script we distinguish between Objects, Functions and Commands –

**Objects** are all the things in the game world, be they visible objects, creatures, NPCs or just sounds.

**Functions** are all the "words" of TES Script that let us either manipulate these objects or let us gather information about them.

**Commands** are those "words" that structure the scripting language, but do not operate on any Game objects – an example is the word "Begin" we used to tell the script editor about the name of our script.

To tell the game which object it is supposed to perform a given function on we can use the "arrow", or "fix" : -> (really just a hyphen and a greater-than sign). You specify the object for the function on the left (we also call this the calling object) and the function to be performed on the right:

Object\_ID->function, [parameters]

A function may or may not have parameters. Parameters could be other object ID's, numbers and in some cases, variables.

What we need for our riddle chest is the `OnActivate` function: this is an informative function that tells us whether the player has "activated" an object in the game world or not. This function returns a value of 1 (which means "true" in programming terms) if the object has been activated, that means the player targets it and presses the "use" button (space, by default). So what we need to do is check if `OnActivate` becomes "true" anytime in the game.

**So edit your script to look like this:**

```
Begin my_first_script
If ( OnActivate == 1 )
    ; here we will enter what happens when the chest is opened
endif
End
```

A couple more things need to be explained here: The "*if*" command is there to check a condition – whenever the expression in the parentheses is "true" the following lines of code will be executed until the "*endif*" command is encountered. The "`==`" checks if an expression (in our case the "`OnActivate`" function) on the left of it is equal to the expression on the right of it (in our case to 1). If you forget the *endif* command after an if command, the editor will complain with an error message. The ";" semicolon denotes a comment – whatever you write behind the semicolon will be ignored when the script is run. If you ever write larger scripts you should learn to love this possibility.

## Writing text and obtaining decisions from the player

Now we want our trapped chest to ask the player a riddle. For this we use the `MessageBox` function that allows us to display some text on the screen and also to display choices that the player can select from. Unfortunately Morrowind has no option to have the player type in the answer to our riddle, so we will have to give multiple choices. The line for that could read:

```
MessageBox "Voiceless it cries, wingless flutters, toothless bites, mouthless mutters. What is it?", "Bat", "Old woman", "Wind", "Wraith"
```

The first text is the text actually displayed in the box, the other texts, separated by commas tell the game to make "buttons", with the text given displayed.

But how do we ensure that the riddle is asked only the first time we try to open the chest and not every time? We now come to a very central point: the use of do-once conditions and state variables. Most of the problems that beginners encounter with scripting for Morrowind have their roots in misunderstanding how the scripts are actually executed and how scripts should accordingly be structured. So let's have a look at this.

## How local scripts are executed

Every script that is attached to an Object or an NPC (local script) is executed *every frame the game displays on screen* while the cell with the object is active (indoors only the cell the NPC is currently in is active, outdoors the PC's cell and all adjacent cells are active). So the *complete script* (not just one line of it) is executed 10-60 times a second or however fast your computer runs the game! It is best to imagine every local script wrapped in a big "while-loop":

```
while (Object is in active Cell)
[Your script code]
endwhile
```

This is the reason why the following script spits out a continuous stream of messages (if attached to an Object or NPC in the same cell as the player). Try it, if you want:

```
Begin Message_script
MessageBox "Thousands of useless Messages"
End Message_script
```

This example is relatively harmless, but imagine what happens if you would use a line of code that adds an item to the players inventory, or places a monster next to him, etc.!

For this reason, "Do Once" constructions are very essential and something you will probably use a lot while scripting for Morrowind. So, let's go on with our tutorial script: we need to declare a variable and use it to make sure the message is only displayed once. **Change the script to the following:**

```
Begin my_first_script
Short controlvar
If ( OnActivate == 1 )
    If ( controlvar == 0)
        MessageBox "Voiceless it cries, wingless flutters, toothless bites, mouthless
mutterers. What is it?", "Bat", "Old woman", "Wind", "Wraith"
        Set controlvar to 1
    endif
endif
End
```

**(Please note that the MessageBox command should be in one line in the editor!)**

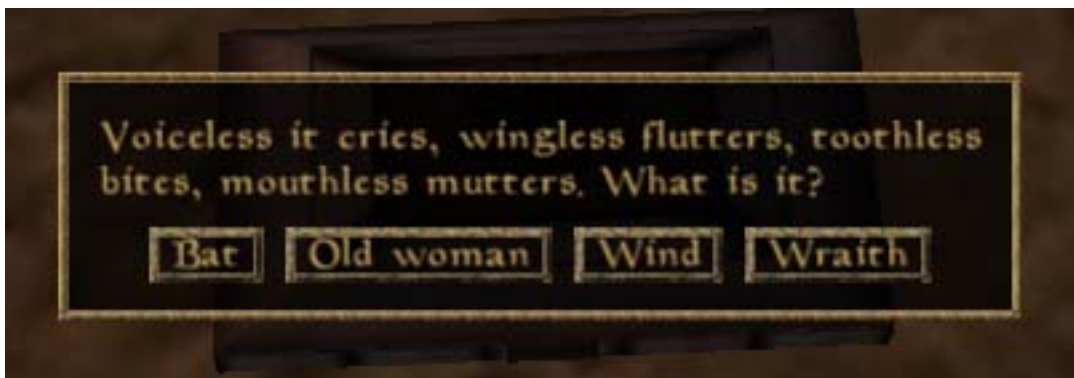
"*Short controlvar*" declares a new variable I called "*controlvar*", of type short. For the moment it's enough to know that this is variable that will contain integers (whole positive or negative numbers). A variable is a "placeholder" that can take on different values. The *if* command we already know, the *set* command is new, but simple enough – it sets our variable that had the value 0 before (all variables start out at zero when declared) to 1. This, in

connection with the *if ( controlvar == 0 )* command provides a do-once condition – the next frame the script is executed after the variable was set to 1 the if condition will be false and the message box will not be displayed again.

Now our script is already capable of being run, so let's test it:

- **Save the script and close the script editor window.**
- **Go to the TES construction set, Object window, select the container tab and open "chest\_small\_01".**
- **Change the ID of the chest to "tutorial\_chest"**
- **In the script dropdown field, select my\_first\_script**
- **Save the object as a new object, save the mod, and quit the construction set. Start Morrowind and load a savegame.**
- **Now bring down the console (usually the ~ key, or whatever you have to the left of the "1" on the main keyboard) and in the console window, type:  
PlaceAtPC tutorial\_chest 1,1,1  
and hit return.**

Take a step back (err, let your player character step back, that is!); you should now have a little chest sitting on the floor right in front of you. Clicking on it should bring up our message, which should look like this:



Clicking on those buttons will just close the messagebox for the moment, and clicking on the chest again, nothing should happen either – which is good, it means our do-once condition works.

**Ok, leave Morrowind and go back to the editor, and load your plugin again.**

We now need to figure out which answer the player selects, and script appropriate reactions for right and wrong answers. The function to test the selected answer is

"GetButtonPressed". This function returns a number depending on which of the buttons of a message box has been clicked on with the mouse. It will return "0" for the first button ("bat" in our example) and 1, 2, 3 etc. for the following buttons, in the order you listed them in the messagebox function. While no answer has been selected, the function will return -1, so we have to take care of that, too.

The "Activate" function will make our chest open, in fact activate will simply trigger the standard action that would usually be performed when you "use" the selected object – e.g. doors would swing open, NPCs would initiate dialogue etc.

The following update to our script also demonstrates how you can use a control variable to force MW to process functions one after the other, although the complete script is processed every frame of the game: simply increment the control variable and test it in a series of if – elseif statements. This is a very safe way of scripting for MW – it may not always be necessary, but it's safe.

**Please edit the script to the following:**

```
Begin my_first_script

Short controlvar
Short button

If ( OnActivate == 1 )
If ( controlvar == 0)
    MessageBox "Voiceless it cries, wingless flutters, toothless bites, mouthless mutters.
What is it?", "Bat", "Old woman", "Wind", "Wraith"
    Set controlvar to 1
elseif controlvar > 1
    activate
endif
endif

if (controlvar == 1)
    set button to GetButtonPressed
    if ( button == -1 )
        return
    elseif ( button == 2)
        MessageBox "The answer was correct"
        Activate
        set controlvar to 2
    else
        MessageBox "The answer was wrong"
        set controlvar to -1
    endif
endif
End
```

Take a look at the part that starts with "*if (controlvar == 1)*". We have set controlvar to 1 as soon as the chest got activated. Now we test for which button is being pressed. We do this by assigning the new variable "*button*" a value returned by *GetButtonPressed*. Since the script is still running, even while the game seemingly pauses to await your decision, we first test if no button has been selected yet – return tells the game engine to stop processing the script for this frame.

Our correct answer was "wind" which corresponds to button number two – if button number two gets pressed, we will tell the player that he gave the right answer, and "activate" will open the chests inventory in the usual way.

All other values of button mean that the player has selected a wrong answer, so we can use the "*else*" command here. In this case we tell the player what a fool he was and the chest is not activated.

Now look at the little addition at the top of the script:

```
If ( OnActivate == 1 )
If ( controlvar == 0)
    MessageBox "Voiceless it cries, wingless flutters, toothless bites, mouthless mutters.
What is it?", "Bat", "Old woman", "Wind", "Wraith"
    Set controlvar to 1
elseif controlvar > 1
    activate
endif
endif
```

This means that, whenever the chest is activated in the future, it will only open if controlvar is greater than 1. Check above: when the player provides the wrong answer in the riddle, controlvar is set to -1, so he will never be able to open the chest. But if he knew the right answer, *controlvar* is set to 2, and from now on the player can open the chest as often as he likes. Save and run your plugin, and test as described above.

## Your first bug

Now, you will probably have noticed that the script does almost – but not quite – what we wanted. After clicking the correct answer, the chest's inventory doesn't open as intended. Now, the logic above seemed fine, so what is wrong? **Let's try the following (change the corresponding part of your script according to the fragment below):**

```
if (controlvar == 1)
    set button to GetButtonPressed
    if ( button == -1 )
        return
    elseif ( button == 2 )
        MessageBox "The answer was correct"
        set controlvar to 2
    else
        MessageBox "The answer was wrong"
        set controlvar to -1
    endif
elseif ( controlvar == 2 )
    Activate
endif
```

See how I moved the activate command to the section that tests for controlvar == 2? This provides a cleaner sequence of events, and as I mentioned above, this can be very important when scripting for Morrowind – always try to avoid doing too many things at once! **Well, run and test it.**

Great, now the inventory opens as we wanted, but what is this? The cursor is real slow, and we can't close the inventory! Look above – *controlvar* was set to two, and remains there, we do not change it again – therefore the game now gets continuous "Activate" commands each time the script is processed (every frame)! That's why we can't close the inventory – it gets reopened immediately. **So change the following part of the script:**

```
elseif ( controlvar == 2 )
    Activate
    Set controlvar to 3
endif
```

**Test again:** now everything works the way we wanted. I hope I have not confused you with the above excursion into the process of debugging, but it is a very important thing to know about – you will constantly have to rethink your scripts and try different ways of doing it to be successful.

What is still missing? The trap effect of course!

## Putting a spell on the player

Our chest will put a curse on the player if he fails to answer the riddle.

**First go to the spellmaking tab in the editor, right click and select "new". Give the spell the ID "Frost\_Curse", name it "Frost Curse" and make it type "curse" then give it a magnitude of e.g. 1-5. It should look like in the picture below.**

| Count | Name        | Type  | Cost | Auto... | PC S... |
|-------|-------------|-------|------|---------|---------|
| 0     | Frost Curse | Curse | 0    | no      | no      |

Spellmaking

ID

Frost Curse

Full Name

Frost Curse

| Effects        | Cost | Range | Area | Duration | Magnitude | Total Cost |
|----------------|------|-------|------|----------|-----------|------------|
| 1 Frost Damage |      | Self  | 0    | 0        | 1 5       |            |
| 0              |      |       | 0    | 0        | 0 0       |            |
| 0              |      |       | 0    | 0        | 0 0       |            |
| 0              |      |       | 0    | 0        | 0 0       |            |
| 0              |      |       | 0    | 0        | 0 0       |            |
| 0              |      |       | 0    | 0        | 0 0       |            |
| 12             |      |       | 0    | 0        | 0 0       |            |
| 19             |      |       | 0    | 0        | 0 0       |            |
| 14             |      |       | 0    | 0        | 0 0       |            |
| 1              |      |       | 0    | 0        | 0 0       |            |
| 4              |      |       | 0    | 0        | 0 0       |            |

Type

Curse

Spell Cost

0

☐ Auto Calculate Cost
 ☐ PC Start Spell
 ☐ Always Succeeds
 

Save

Cancel

☐ Blocked

Now, we need to put this curse on the player. For this we use the *AddSpell* function. After some time we will remove the curse again using the *RemoveSpell* function, for this we need to make a timer. Edit your script again:

```

Begin my_first_script

Short controlvar
Short button
Float timer

If ( OnActivate == 1 )
If ( controlvar == 0 )
    MessageBox "Voiceless it cries, wingless flutters, toothless bites, mouthless mutters.
What is it?", "Bat", "Old woman", "Wind", "Wraith"
    Set controlvar to 1
elseif controlvar > 1
    activate
endif
endif

if (controlvar == 1)
    set button to GetButtonPressed
    if ( button == -1 )
        return
    elseif ( button == 2)
        MessageBox "The answer was correct"
        Activate
        set controlvar to 2
    else
        MessageBox "The answer was wrong"
        Player->AddSpell, "Frost_Curse"
        set controlvar to -1
    Endif
elseif ( controlvar == 2 )
    Activate
    Set controlvar to 3
elseif ( controlvar == -1 )

```

```

        Set timer to ( timer + GetSecondsPassed )
        if timer > 10
            Player->RemoveSpell, "Frost_Curse"
            set controlvar to -2
        endif
    endif
End

```

Let's go over this. Player->AddSpell, "Frost\_Curse" puts the curse we created earlier on the player. Note how we need to use "Player->" to make sure the effect really targets the player. Otherwise we would curse the chest (which is the default object, because the script is attached to it), which wouldn't make a lot of sense...

This bit:

```

Float timer
Set timer to ( timer + GetSecondsPassed )
    if timer > 10

```

...that is how you make a timer in Morrowind. *GetSecondsPassed* is a function that returns the time in seconds that has passed since the last frame. Since this is usually a fraction of a second (as the script gets called every frame), it is only natural that we need a float variable for this purpose – a variable that can store numbers with decimals. So when the timer has been running for 10 seconds we remove the curse again, and make sure we do this only once:

```

        Player->RemoveSpell, "Frost_Curse"
        set controlvar to -2

```

Ok, save and test your mod. Works fine now, doesn't it? Well, almost. Try the following: let yourself be cursed, and then open your inventory. Wait. See how the curse terminates after some time, without hurting you? Of course: the script is still running, but spell effects are only calculated while in game, not while you are in the menu. We don't want the player to get off the hook so easily, so we need to put something in our script that stops it from processing when we are in menu mode. Luckily there is the *MenuMode* function, which returns 1 when you enter the menu. So we can put this at the start of our script:

```

If ( MenuMode == 1 )
    Return
Endif

```

Remember, return tells the game to stop processing the script for this frame.

Ok, now we have our final working script. Congratulations! If you want, experiment a little more with this script: Put the chest into a location in the game and lock it. Then try unlocking it (Unlock function) with the script in addition to activating it. Try adding a sound, when the riddle is successfully solved (e.g. PlaySound3D, "skillraise"). Try using the "Cast" function instead of "AddSpell". Previous users of the Tutorial have spotted a potential bug in it: what happens if the player leaves the area with the scripted chest before the curse is removed again? How would you fix this?

This is the final script:

```
Begin my_first_script

Short controlvar
Short button
Float timer

If ( MenuMode == 1 )
    Return
Endif

If ( OnActivate == 1 )
If ( controlvar == 0 )
    MessageBox "Voiceless it cries, wingless flutters, toothless bites, mouthless mutters.
What is it?", "Bat", "Old woman", "Wind", "Wraith"
    Set controlvar to 1
elseif ( controlvar > 1 )
    activate
endif
endif

if ( controlvar == 1 )
    set button to GetButtonPressed
    if ( button == -1 )
        return
    elseif ( button == 2 )
        MessageBox "The answer was correct"
        Activate
        set controlvar to 2
    else
        MessageBox "The answer was wrong"
        Player->AddSpell, "Frost_Curse"
        set controlvar to -1
    Endif
elseif ( controlvar == 2 )
    Activate
    Set controlvar to 3
elseif ( controlvar == -1 )
    Set timer to ( timer + GetSecondsPassed )
    if timer > 10
        Player->RemoveSpell, "Frost_Curse"
        set controlvar to -2
    endif
endif

End
```

## How to learn more

After this tutorial you may ask yourself how to continue with learning how to script. A good way is to look at the example scripts in this guide or at scripts that are already in the game (either from Bethesda or from mods). Try to find a script that is similar to what you want to do and then copy the script and change it to fit your needs. Read the general information below and the descriptions of the functions you may need to do what you have planned. The ordering of the functions into thematic groups should help you to find the right ones. Finally, the official forums are a great place to find information (use the search function) or to get help on a specific problem. The rest is practicing, practicing, practicing 😊

# General Information: Scripts, Commands and Syntax

## *Types of scripts*

### Local scripts

Any script that is running on an object or Actor in the game (assigned in the script-dropdown field of the object or Actors object window) is a local script. Local scripts are only active if the cell is loaded – this is the current interior cell, or the current and all directly neighboring exterior cells. When the object is outside of this range the script is not running, but the local variables are saved. You cannot stop a local script using Stopscript.

### Global scripts

Any script that is not attached to any object is a global script, and is by default not executed until you call it (see below). Note that there is no default object for a global script to work on, so objects must always be specified: while the following will work in a local script attached to an NPC:

```
AITravel 1150, 8899, 1110
```

You will have to specify the NPC in a global script:

```
"NPC_ID"->AITravel 1150, 8899, 1110 ;NPC_ID is the ID, the unique identifier for  
;each object in the editor
```

Global scripts are active all the time once they have been activated and until they are specifically terminated. Thus, once activated, they will be processed every frame as described for locals scripts above. That is why they should be used with caution, as too many, or too complicated global scripts can easily slow the game down a lot.

The command to start a non-active script is:

```
StartScript, "Script ID"
```

With Tribunal and Bloodmoon you also have the option to make a script start automatically with the game. In the TESCS under the menu Gameplay/Edit starting Scripts/ you can add any script to the list of automatically started scripts.

The function to terminate a global script is:

```
StopScript, "Script ID"
```

Variables local to a global script will be saved temporarily when the script is stopped and restarted. In order to ensure that variables are always saved, the script must be run at least once every load session. If you need to be sure the variables are reset, you should reset them yourself: don't rely on it happening automatically.

A simple way to ensure variables are saved is to use a startscript (only available with expansions); so for example the startscript might look something like this:

```
begin KeepVarsScript  
  
if ( ScriptRunning, MyScript == 0 )  
    set MyScript.KeepVars to 1  
    StartScript MyScript  
endif
```

```
StopScript KeepVarsScript  
end
```

...and the script "MyScript" might look like this:

```
begin MyScript  
short KeepVars  
if ( KeepVars == 1 )  
    set KeepVars to 0  
    StopScript MyScript  
    return  
endif  
;main body of script goes here  
end
```

It is possible to use the StartScript function to run global scripts that are tied to an object or Actor. These are called "targeted scripts".

```
"Object_ID"->StartScript "Script_ ID"
```

These scripts resemble both local scripts (in that the functions called always default to the object or Actor the script targets) and global scripts (in that they are always running and can be terminated with StopScript).

**Note:** read more on the special case of "targeted scripts" in the Tips and Tricks section.

## Syntax

There are some unique features in TES script that should be explained before we go into more detail.

### Beginning and ending scripts

**Begin Script\_ID**

**End**

**End Script\_ID**

Every script must have the Begin and End tags. The specified name will also be the ID you will reference the script by (be it from other scripts or inside the TES CS object windows). A script may start out with comments, but the first line of real code must be "Begin xxxxxxxxxx". As with other objects, it is recommended that you give your scripts a unique tag. I usually use GBG\_Scriptname. This ensures that your scripts are easily identified, they all are neatly listed in one block in the scripts list, and there is little danger of conflicts with other mods using the same name. Using leading underscores e.g. \_Scriptname is not recommended.

If you want your scripts to appear at the top of the list of scripts, put a 1 in front, for example, 1YAC\_ScriptName

### General syntax for functions:

Functions are not case sensitive, but using the case sensitive form as suggested by Bethesda (e.g. GetSpellEffects instead of getspelleffects) makes scripts easier to read, so it's recommended to keep it that way.

```
"Object_ID"->Function, [parameters]
```

This is the format for all functions that act on or refer to a specifiable object in the game world. The "arrow" or "fix" designates which object a function will be performed on. The "object\_ID" is the unique ID that is given to each object in the editor (usually the first field in any object window). You need this ID, not the Name! If you call a function without a designated Object-ID, the function will be performed on the default object, which is the object the script is attached to.

This is not to say that there might not be another object referenced as a parameter:

```
"Object_ID1"->Function, "Object_ID2"
```

### Notes:

Functions with a "fix" will only compile if the object has already been placed into the game world in the editor (with at least one reference), otherwise the script will not compile. If you compile the script and then delete all references, the game will give errors on load and CTD.

More than one function may be used in a *set* expression, but the functions always apply to the same reference. For example:

```
set SomeVar to ( player->GetStrength ) + ( player->GetEndurance )
```

and

```
set SomeVar to ( player->GetStrength ) + ( GetEndurance )
```

will both have the same effect, and specifying different references will not work.

Referencing non-unique items with the "fix" will usually perform the function only on the first reference of the object! So using this in a global script:

```
"cliff racer"->ModCurrentHealth -1000
```

will not have the desired effect, it will only kill one of the annoying bastards. However attaching a script to the creature with just

```
ModCurrentHealth -1000
```

Would do the trick, because every reference of the cliff racer will have the script running, and apply the function to itself. Note that this does not apply to all functions (e.g. SetHealth in the above example would affect all references the player has not yet encountered).

A number of functions refer only to the player or not to an object at all, and are therefore using the fix is meaningless or may produce errors. E.g.:

```
If ( GetPCRank == 0 )
If ( CellChanged == 0 )
FadeOut, 2
```

## General syntax: Commas, parentheses and spaces

TES Script is not too picky about syntax. Case mostly doesn't matter, commas can be left out, and spaces are mostly ignored. Nevertheless I would advise adhering to the following principles:

- Either avoid commas, or always use commas: inconsistent usage can cause problems.
- If the ID contains a space or begins with an underscore, you must use quotation marks: "Object ID" or "\_Object\_ID". Better to avoid spaces altogether: Object\_ID
- Get used to **always** leaving spaces around parentheses and operators, sometimes it seems to cause problems if you don't: if ( variable == 1 ), not: if (variable==1). While this doesn't matter most of the time it generates weird and almost untraceable errors sometimes, so you are much better off always leaving a space.
- The fix (->) is a little more complicated. If IDs are contained in quotes, you should **not** leave spaces around the fix:

```
"Sirollus Saccus"->GetItemCount netch_leather_greaves
```

The above will work, but spaces around the fix would cause problems. (Thanks to Simpleton and DinkumThinkum for this info.) However, since it has also been reported that a *lack* of quotation marks can cause problems in combination with spaces round the fix, I'm going to recommend that you don't leave spaces round the fix at all (should be fine in all cases).

## Comments

Comments are marked by a semicolon ;. Comments can be added in their own lines or behind lines with code

```
; enter sneak mode
Fargoth->ForceSneak
Fargoth->AiTravel -11468.595, -71511.531, 173.728 ; goes to tree
```

## Indentation / using tabstops

For your own sake, use proper indentation (tabstops) for if-elseif constructs – makes it much easier to keep track of them, so you don't forget an endif at the end. In the tips and tricks section you will find a link to an external EMACS editor mode for TES script that will do automatic indentation.

```
If ( variable1 )
    If ( variable2 )
        [do something]
    endif
endif
```

is better than

```
If ( variable1 )  
If ( variable2 )  
[do something]  
endif  
endif
```

## Variables

### Types of variables

There are three types of variables in the TES script language: short, long and float. According to the manual these cover the following data ranges:

|       |                                                        |
|-------|--------------------------------------------------------|
| Short | -32,768 to 32,767 (signed integer)                     |
| Long  | -2,147,483,648 to 2,147,483,647 ( long signed integer) |
| Float | 3.4E +/- 38 (float, 7 digits)                          |

Apparently the boundaries for Long given here are only partly correct, in the TES-CS you can assign a maximum value of 2147483520 (Forum info / Argent).

Theoretically there should be string variables too, but to my knowledge these are not implemented. Unfortunately there are also no data types to store Object\_Id's, which limits the power of the scripting language to a certain extent.

Variables can be grouped into *local* variables (valid only in the script that declares them) and *global* variables (valid in every script).

**Note:** A long global is effectively a float! In a script, local to that script a long will have a full 32 bits of precision. But used as a global, the number of bits of precision drops to 24, as when it exists globally a long is a float. Mental Elf discovered this when attempting to use "bit packing" to get 32 flags into a global long (forum info / mental elf).

#### A Note about Floats:

Floating point numbers in a computer are stored in a format similar to that used for scientific or engineering notation. The number is rounded off to a fixed number of significant digits, any leading or trailing zeros are dropped, and an exponent is added to indicate the correct location of the decimal or binary point.

The number 2385901045 would be stored similar to this:

2.385901x10<sup>9</sup>

In fitting the number into a fixed number of bytes, we have lost the end 3 digits, so the number is no longer quite what you set it to.

If you set a float to a small number, like 5.5, it would still be stored as 5.5, because as it doesn't take up much room, no numbers need to be cut off.

The advantage of this format is that a wide range of values can be expressed using a relatively small number of digits or bytes. The disadvantage is that the rounding off means floating point numbers don't have the precision of an integer, so they shouldn't be used when you need to make exact comparisons or keep track of exact counts.

For example, a floating point variable is fine if you just need to check if a distance is greater or less than a certain value. But if you need to check if the player has specific number of some item in their inventory, then you want to use a long or short integer variable for the count. (Forum info / DinkumThinkum)

An example by BungaDunga shows that when doing math with floats, the answer isn't always what you expect:

```
Float num1
Float num2
Float num3

Set num1 to ( 1 / 3 )
;num1 is now 0.3333333

Set num2 to 3
;num2 is now 3

set num3 to num1 * num2
;num3 is now 0.9999999, rather than what you would expect, 1.

if ( num3 == 1 )
    ; Never happens.
endif
```

A lot of functions return float values (e.g. GetDistance, GetScale, GetSecondsPassed...): the same thing applies. Test a range, not an exact value! E.g.:

```
if ( ( GetPos x ) == 500 )
    ;will be false if it's 499.9999, or 500.0001, etc
endif
```

but this will work:

```
if ( ( GetPos x ) > 499.5 )
    if ( ( GetPos x ) < 500.5 )
        ;it's close enough for me
    endif
endif
```

Another common mistake is to check if the GameHour is an exact number. As the Gamehour variable is increased every frame, and every frame is a slightly different length, the Gamehour variable soon ends up with a value like this, 10.12853. So never test if the gamehour is exactly equal to a value, as it is very unlikely to happen, test if it is greater than or equal to a value.

## Local variables

Local variables these have to be declared in the script:

```
Float floatvarname
Short shortvarname
Long longvarname
```

Local variables are unique to a specific instance of a local script. This means that the local variables in multiple objects with the same script do not influence each other. The names you use for variables are pretty much up to you as long as they start with a letter, but you have to avoid using function names (this will result in errors during runtime) and reserved characters (e.g. - + / \* = " ) ( etc.) which will result in compiler errors. E.g. "variable-1" will not work as a variable name. Underscores as in "my\_variable" are ok, but avoid leading underscores. A dot has a reserved meaning as well (see "*Referencing variables in other local scripts*" below).

## Global variables

To declare a global variable go to the Gameplay menu and select Globals. Right click for "new", name it, and set the type and starting value for the new global variable, if one is needed. By default it will be 0. Global variables are very useful for involved quests when you need to keep track of things over an extended time and space. They are also a simple way to share information between different scripts

**Note:** if you declare a local variable with the same name as a global variable, the global variable will become invisible for this script. Do NOT declare a global variable as a local!

## Referencing variables on other objects and scripts

### Set ... to

If a unique object has a local script running on it you can change variables from outside the script in the following way:

```
Set MyObject.variable to 100  
OR  
Set MyObject.variable to local_variable
```

This method changes a local variable in the object's script. The object must have a script on it for this to be valid. The object does not have to be in an active cell when setting the variable, but it will only work if the cell containing the target object/(script) has previously loaded - (Cyran0). **Note:** The scripting system looks at the first object in the database, thus you should only reference objects that are unique (exist only once).

Note that the reverse does **not** work:

```
Set local_variable to MyObject.variable ;this doesn't work!
```

Use a global variable to transfer information in this way, or set local\_variable from the **other** script using the syntax above.

```
if ( anotherobject.x > 0 )
```

apparently works.

Furthermore I realized only recently that this syntax also works for global scripts:

```
set Global_script_name.variable to 1
```

This is useful to avoid using more global variables than necessary or also as a console command to debug global scripts.

**Note:** If your object or global script starts with a number, you need to put quotes around it.

```
"11NPC01".RemoteVar ;---Good  
"11NPC01.RemoteVar" ;---BAD  
11NPC01.RemoteVar ;---BAD
```

### A caution when setting variables from outside the script (DinkumThinkum):

*If you recompile the target script (i.e., the local script on 'MyObject'), it's a good idea to also recompile any scripts that reference variables in that script. Reason: if the target 'variable' winds up in a different position in the target script's list of variables, then any scripts trying to set that variable will break if they're not recompiled.*

*I.e., if 'variable' is the 14th variable declared in the local script on 'MyObject', and you add a new variable ahead of it so that 'variable' is now the 15th variable declared, other scripts will need to be recompiled in order to find 'variable' in its new position, otherwise the script will end up changing the wrong variable, leading to very strange bugs.*

*One way to reduce the chances of this tripping you up is declare variables referenced by other scripts first, before other variables that aren't referenced externally. Then just be careful to only add or remove variables after the block of externally referenced variables. However (to be safe), it's still a good idea to recompile scripts that reference variables in other scripts any time you've added or removed variables in the other scripts.*

## Using variables in functions

Unfortunately it is one of the limitations of TES script that only certain functions accept variables as parameters, which poses some definite limits. The type of arguments functions accept is indicated in the list of functions below.

**Note:** For some functions where both a Get-Function and a Set-Function exist, a workaround can be constructed by using the while function (see below).

## Operators / mathematical calculations

You can use the standard operators to do calculations in scripts:

|                 |   |
|-----------------|---|
| Addition:       | + |
| Subtraction:    | - |
| Multiplication: | * |
| Division:       | / |

The syntax is as follows:

**Set result\_var to (var\_a + var\_b)**

Instead of variables, literal values are also allowed. I assume that standard operator precedence applies ( \* and / are calculated before +/- ). Since it has not been properly tested I usually always use parentheses to be on the safe side. You can use parentheses according to standard mathematical rules:

```
set ln to ( ln + ( k10 * math_ln10 ) + ( k2 * math_ln2 ) )
```

A warning: There are different opinions on the forums on the use of several operators in one line. Some people report a lot of problems with this, I myself have successfully used at least four operators and variables in a single line. There appears to be an issue with very long additions (e.g. adding up more than 20 variables in one line of code) that causes a mod to crash the game upon loading. If this happens, split the calculations to several lines.

### Notes by DinkumThinkum:

*Eleven variables in a single set statement will cause the following error when loading the game, followed by a crash to desktop:*

**"Need more room for zero pointers in Script::ReplaceGlobalsInData"**

*as soon as you click the button to acknowledge the error, CTD. (That script name wasn't any script of mine; it's something internal to the game.)*

*Six variables in a Set statement work fine. Don't know the maximum number, but it's obviously at least six and definitely less than eleven.*

There isn't much in the way of dedicated mathematical functions in TES script. There is the 'Random' function and Tribunal added the 'GetSquareRoot' function (see below). If you need more complex functions, I suggest downloading Soralis' Math Mod (available from Planet Elder Scrolls). It's a collection of scripts that allow you to do complex calculations. Here is a short excerpt from the readme to give you an idea:

*"This mod adds the ability to use various math functions within Morrowind's scripts. Specifically, these are the scripts that are added:"*

| Name          | Check/Done | Inputs                 | Outputs                      | Accuracy |
|---------------|------------|------------------------|------------------------------|----------|
| MathScripts   | N/A        | N/A                    | N/A                          | N/A      |
| MathConstants | N/A        | N/A                    | N/A                          | N/A      |
| SquareRoot    | 1          | math_sqrt              | math_result, math_imag       | 7        |
| SineScript    | 2          | math_angle             | math_sin, math_cos, math_tan | 7        |
| ArcsineScript | 3          | math_arc               | math_sin, math_cos           | 6-7      |
| NaturalLog    | 4          | math_log               | math_result, math_imag       | 4-5      |
| LogScript     | 5          | math_log, math_base    | math_result, math_imag       | 3-4      |
| intPower      | 6          | math_value, math_power | math_result                  | 7        |
| intRoot       | 7          | math_value, math_root  | math_result, math_imag       | 6-7      |
| Modulus       | 8          | math_value, math_mod   | math_result                  | 6-7      |
| Antiln        | 9          | math_log               | math_result                  | 4-5      |
| Antilog       | 10         | math_log, math_base    | math_result                  | 2-3      |
| AbsoluteValue | 11*        | math_abs               | math_abs                     | 7        |
| PowerScript   | 12         | math_value, math_power | math_result, math_imag       | 2-3      |

Unfortunately many of these functions are rather slow, and not suited for real time calculations. For sine and cosine in particular, check out JDGBOLT's script in the tips and tricks section, which uses pre-calculated values for a very rapid calculation. You can also find more information on UESP Wiki.

MWSE (Morrowind Script Extender) also adds trig functions to Morrowind.

## Testing conditions

### Use of if... elseif conditions

```
If (condition)
Elseif (condition)
Else
Endif
```

Much of TES scripting relies on the use of *if... elseif* conditions and variations thereof. It is very important to fully understand these, and the conditions that can be used in them to test conditions of the game world to trigger events.

In general a condition is "true" when it has the value of 1 (or simply "not 0" (e.g. -1 is "true" in TES Script too) and "false" when the value is 0. There are thus certain functions in the game that return "true" under certain conditions. For example see the GetAIPackageDone function further down. It returns "true" (= 1) for one frame when an AIPackage has finished. All lines of code between the if-statement and endif (which concludes a so-called "if block") will be executed only when the condition is true.

Thus:

```
If ( GetAIPackageDone )  
    ;[do something here]  
endif
```

will be only executed when the GetAIPackageDone function currently returns the value 1.

In addition to just a value, a condition can also be a more explicit test of conditions. The conditions, such as "A equals B" or "A is greater than B" will be evaluated, and the whole expression will be true (equals 1) when the condition is fulfilled. E.g.:

```
if ( GetAIPackageDone == 1 )  
    ;[do something here]  
endif
```

Is equivalent to the example above. This second syntax tests a condition and returns "true" if the condition is met. You can check for the following conditions:

| Verbose                    | Conditional operator |
|----------------------------|----------------------|
| "Equal to":                | <b>==</b>            |
| "not equal to"             | <b>!=</b>            |
| "Greater than"             | <b>&gt;</b>          |
| "Smaller than"             | <b>&lt;</b>          |
| "Greater than or equal to" | <b>&gt;=</b>         |
| "Smaller than or equal to" | <b>&lt;=</b>         |

If you have several independent if blocks each will be evaluated separately. If, e.g. the following if blocks are in your script:

```
if ( timer >= 2 )  
    ;[Block A,do something here]  
endif  
if ( timer <= 3 )  
    ;[Block B, do something else here]  
endif
```

Both will be true and the code in the blocks will be executed when timer is between 2 and 3. You can nest if-blocks if you want to make sure that two conditions are fulfilled simultaneously:

```
if ( timer >= 2 )  
    if ( controlvar == 0 )  
        ;[do something here]  
    endif  
endif
```

The code in the inner if block will only be executed when both conditions (timer >= 2 and controlvar == 0 ) are fulfilled at the same time.

For more elaborate constructions you can use Else and Elseif. Elseif tests for a separate condition if the preceding condition has failed (but not if the previous condition was fulfilled)

```
if ( timer >= 2 )  
    ;[Block A,do something here]  
elseif ( timer <= 3 )  
    ; [Block B, do something else here]  
endif
```

unlike the example above, both blocks can not be true at the same time now. Either timer is >= 2, then block A gets executed, or timer is not >= 2 but <= 3, meaning effectively timer is <

2, then block B gets executed. In all other cases, neither block will be executed. You can have several `elseif`s behind each other:

```
if ( counter == 1 )
    ;[Block A,do something here]
elseif ( counter == 2 )
    ; [Block B, do something else here]
elseif ( counter == 3 )
    ; [Block C, do something else here]
endif
```

An `else` creates a "default condition". The code following `else` will be executed if all previously tested conditions are false:

```
If ( foo_var == 1 )
    [Block A: do something]
elseif ( foo_var == 2 )
    [Block B: do something else]
else
    [Block C: do something completely different]
endif
```

In this example Block C will be executed if `foo_var` is neither 1 nor 2.

### Notes and cautions:

In my experience it is safest to use *elseif* instead of multiple separate *if* statements if you test different states of one variable.

Be careful using a function in a conditional ('If' statement); simple functions seem to be OK, but ones with numeric parameters don't seem to be reliable.

For example:

"If ( NewType != OldType )" worked correctly, but "If ( ( Player->GetArmorType 0 ) != OldType )" always registered as 'Not equal', even if the variable 'OldType' had the same value as the armor type that should have been returned by 'Get ArmorType 0' function.

So use "Set NewType to ( Player->GetArmorType 0 )", followed by a separate 'If' statement to compare 'NewType' to 'OldType'. -(DinkumThinkum)

## While conditions

**While ( condition )**

; things to do

**EndWhile**

The `while` command differs from the `if` command in that it is repeated within one frame until the condition is fulfilled. This is best explained with an example:

```
Short desiredAmnt

SetStrength 0
while( GetStrength < desiredAmnt ) ; non-literal value to match
    modStrength 1
endwhile
```

This will set strength to the value in variable `desiredAmnt` after one frame. The following script however would need an undetermined amount of frames to do this, because the `if` condition is only called once each frame:

```
if(getStrength < desiredAmnt) ; non-literal value to match
    modStrength 1
endif
```

On the other hand the first example can potentially cause "freezing" (if the value would be very high) while the second won't.

**Note that this is a workaround for some functions to the problem of functions not accepting non-literal values (variables) as arguments.**

## Constructing Boolean operations in TES Script

Unfortunately there are no Boolean operators (AND, OR, NOT, XOR, ...) in the scripting language. Thus you need to construct these yourself using if... elseif structures.

Instead of AND:

```
if ( variable1 AND variable2 ); does not exist
    [do something]
endif
```

you have to use:

```
If ( variable1 )
    If ( variable2 )
        [do something]
    endif
endif
```

For OR constructs:

```
if ( variable1 OR variable2 )
    [do this]
endif
```

you can use elseif constructions:

```
If ( variable1 )
    [do this]
elseif ( variable2 )
    [do this]
endif
```

# List of TES-script functions

## Explanation of the format

First I will list the function and the arguments it takes:

**[no fix] Code "string", arg\_enum, arg\_float, [optional] (returns short)**

[no fix] Indicates this function can never be used with a "fix" meaning it can not be called by a specified actor. Functions without this tag can be called by an actor, an object, or both.

Code: Name of the function

Arguments of the function: "string" indicates a literal string, such as an object ID. arg\_enum indicates a literal value (no variables taken) var\_float indicates a variable of the specified type (in this case float). Brackets [] indicate optional parameters. (returns short) or (returns float) indicate that the function returns a value and of what type the value is. I will use the designation (returns Boolean/short) to indicate a function that returns either 1 or 0 (a Boolean variable although the game strictly speaking still uses a float)

Examples of usage follow in italics and are indented:

*Code "ID", var\_enum, var\_float*

Example scripts are set in frames:

```
Begin script
[Script functions]
End script
```

From the 8<sup>th</sup> edition onward the functions added with Tribunal and Bloodmoon have been moved into the suitable sections of the reference section. They are now marked with:



for Tribunal functions and



for Bloodmoon functions.

To use these functions the respective expansion must be installed (but not necessarily active). Bloodmoon (and GOTY edition) incorporates all functions from Tribunal as well.

## Working with objects

### Working with inventory items

#### Adding and removing items from the inventory

**AddItem, "ObjectID", count\_enum**  
**RemoveItem, "ObjectID", count\_enum**

```
Actor->AddItem, "item_ID", 1  
Container->RemoveItem, "itme_ID", 5
```

These functions are simple enough, adding or removing items from the player's or any other inventory, including containers. A *RemoveItem* call will remove the item from the inventory, it will "vanish".

#### Notes on AddItem (by DinkumThinkum):

If you add items to an NPC's inventory when the inventory screen is already open, the display won't be updated with the new items, unless the player manually adds or removes displayed items (which forces the game to refresh the inventory display).

The early versions of my Potion Saver coded used 'If MenuMode == 1' to trigger putting potions back into the companion's inventory. This worked fine when the companion was alive and conscious, since the script would add the potions while in the dialogue window, *before* the companion share inventory window was opened.

However, if the companion was dead or unconscious, clicking on the companion would open the inventory window immediately, and the potions would be added after the window was already open. End result, the potions wouldn't show up in the inventory window.

To avoid the problem, add items to an NPC's inventory as soon as they die, as soon as they fall unconscious, or when the game goes into Menu Mode (check Menu Mode *after* checking death and unconsciousness). That way, the items will be added before the inventory window opens.

Here's the code I used for this in the Potion Saver:

```
If ( GetHealth < 1 )  
Set DTNPS_HandlePotions to 0  
ElseIf ( GetFatigue < 1 )  
Set DTNPS_HandlePotions to 0  
ElseIf ( MenuMode == 1 )  
Set DTNPS_HandlePotions to 0  
Else  
Set DTNPS_HandlePotions to 1  
EndIf
```

#### Usage with containers:

If a container has been emptied by the player (in game), AddItem will no longer work. One workaround is to replace the container with a new one each time it's accessed by the player, but this may not be practical as there's no way to detect when the container is empty. Aragon suggested a solution:

*The workaround is to attach a script to the container that always adds and removes a dummy item in the same frame that the player activated it. So, my ingredients chest has a script that looks like:*

```

; Open the chest
Activate

; Always add and remove a dummy (this hack keeps the "AddItem" calls working)
AddItem, "misc_de_cloth10", 1
RemoveItem, "misc_de_cloth10", 1

```

*Furthermore, we make the chest with "references persist". Now we can reference it from an ingredients sorter using code like:*

```

set count to ( player->GetItemCount, "ingred_dreugh_wax_01" )
while ( count > 0 )
    set count to ( count - 1 )
    player->RemoveItem, "ingred_dreugh_wax_01", 1
    "_tm_ingredients_chest"->AddItem, "ingred_dreugh_wax_01", 1
endwhile

```

Another (probably better) workaround suggested by Enmesharra: Add a non-carryable light with no world or inventory art associated with it to the container. It won't show in inventory and thus the container can't be emptied (not tested).

### Notes on RemoveItem:

Removing items that are not present in the inventory does not crash the game, but the 'RemoveItem' function will subtract the removed item's weight from the character's encumbrance, EVEN IF the item is NOT in the character's inventory. So if a script uses 'RemoveItem' to remove a 4 lb. item that the character doesn't have, the character's encumbrance will wind up 4 lbs. lower than it should be. The workaround for this bug is to always check for the presence of an item before using 'RemoveItem' to delete it (Thanks DinkumThinkum for this info).

Don't remove an item that is executing a script, from the item's own script: that will crash the game. See the example script.

A workaround for this is to use a separate global or local script to remove the item. However, if the player has **two or more** copies of an object with an attached script in their inventory, using RemoveItem on that Object ID will frequently corrupt data for one of the remaining copies (you may e.g. see corrupted health or count data). Equipping or using that corrupt object may cause the game to crash (Forum info / DinkumThinkum).

To avoid the two known RemoveItem bugs:

1. If the object is not scripted, use GetItemCount to be sure the player has *at least* 1.
2. If the object is scripted, use GetItemCount to be sure the player has *exactly* 1, or make sure the object is unique.

The drop function *does not* have this bug. It does, however, cause doubling, if used with OnPCEquip / SkipEquip, which can be fixed by adding and removing any item as described for these functions.

### Notes on using Additem / Removeitem in dialogue:

**These functions can accept global variables, but only in dialogue results**, and only if you don't set the global variable in the same dialog result (Forum info / Argent; According to Argent, the maximum amount he has been able to add using AddItem, var was 65534 (using a long var=2147483520)). In addition to Argent's info about AddItem/RemoveItem accepting variables in dialog action boxes: make sure that the variable in question doesn't change in the same box. Say, var\_a equals 3 at the moment when the line is selected in dialogue. If the result field looks like this:

```
set var_a to var_a + 10 ;
AddItem gold_001, var_a
```

three drakes will be added to NPC's inventory, and not thirteen. If such need arises, put all calculating operators into one dialogue response's result box, add "choice" operator, and process the AddItem function in the following response. Also, even with long variables, I couldn't correctly add values above 32767 ( $2^{15}$ ). (tested on pure Morrowind without expansions) (Forum info / Kir).

### Example script:

The following script was discussed on the forums (sorry don't remember who first made it). It was supposed to ask the player if he wanted to recycle an item when it was equipped, and then replace that item with the "recycled" version.

```
Begin scr_thing

short button
short OnPcEquip
short state

if ( MenuMode == 1 )
    return
endif

if ( OnPcEquip == 1 )      ; when the item is equipped
    set state to 1
    set OnPcEquip to 0      ; do this once per equip event...
endif

if ( state == 1 )
    MessageBox "recycle?" , "yes", "no"
    set state to 2
elseif ( state == 2 )
    set button to GetButtonPressed
    if ( button == 0 )
        PlaySound "mysticism cast"
        player->RemoveItem "item_a", 1 ;this line crashes the game!!!
        player->AddItem "item_b", 1
        set state to 0
    elseif ( button == 1 )
        set state to 0      ; once done, reset everything
    endif
endif

end
```

It worked nice enough without the *RemoveItem* function in it, but with that line it would crash. The reason was that this script was *attached* to item\_a, and thus the running script would be removed with it, which apparently crashes the game. So the above idea had to be handled via a global script.

### Dropping items to the floor

**Drop, "ObjectID", count\_enum**

```
"Actor_ID"->Drop, "ITEM_ID", 1
```

This function is supposed to drop the item from the inventory "to the calling Actor's feet". This seems to work correctly only for the player character, dropping the item to his feet. When used on NPC's, the items are removed correctly from the NPC, but still dropped at the player's feet.

An interesting note on this is that if they really have that item, they will drop it, with charges and condition intact. If they don't really have that item, a new instance will be created. If the player drops an item he doesn't have, his weight will still be reduced by the weight of that item - same as the encumbrance bug for RemoveItem (Forum info / DinkumThinkum).

Note that Drop will only work in active cells: if it is used on an NPC in an inactive cell, nothing will happen. -(DinkumThinkum)

**Examples** of use are found in Bethesda's SlaveScript.

## Monitoring inventory activities: Adding, dropping items and using soul gems

**[no fix] OnPCAdd (is local short variable)**

```
Short OnPCAdd  
If ( OnPCAdd == 1 )
```

This variable is set to 1 when the PC added the object to inventory. Must be reset manually for multiple use (set OnPCAdd to 0 ).

**Example:** This is an excerpt from the script attached to Fargo's ring, that gives you the magic menu during character creation:

```
if ( OnPCAdd == 1 ); Player has added the item to the inventory  
    if ( State == 0 )  
        EnableMagicMenu  
        MessageBox "You now have a Magic Menu, where you can see all your powers,  
spells, and magic items." "Ok"  
        set state to 10  
        return  
    endif  
endif
```

**[no fix] OnPCDrop (is local short variable)**

```
Short OnPCDrop  
If ( OnPCDrop == 1 )
```

This variable gets set to one when the PC drops the object. Must be reset manually for multiple use (set OnPCDrop to 0 ).

**[no fix] OnPCSoulGemUse (is local short variable)**

```
Short OnPCSoulGemUse  
If ( OnPCSoulGemUse == 1 )
```

This gets set to one if the calling Object is a soulgem and it has been used in either recharging or item making. Must be reset manually for multiple use (set OnPCSoulGemUse to 0 ).

Example: this is how Azura's star becomes an inexhaustible soulgem:

```
begin AzurasStarScript  
;this is for Azura's Star. A never ending soulgem item. Mucho good. Kids love it.  
  
short OnPCSoulGemUse  
  
;they use it, so give them a new one  
if ( OnPCSoulGemUse == 1 )  
    Player->additem, "Misc_soulgem_Azura" 1  
endif  
  
end
```

## Force-equipping an Item

**Equip, "Object\_ID" [count\_enum]**

```
"Actor_ID"->Equip, "iron dagger"
```

(Also see OnPCEquip function below)

**Pre-Tribunal: Partly Broken.** This could have been an immensely useful function. Unfortunately, most of the potential uses do not work: You can NOT autoequip anything on the Player. You can NOT force the Actors to equip weapons or armor with this (this is completely governed by their skills with armor or weapons). You can NOT make "non-removable" items, like cursed armor, etc. You CAN make Actors swallow potions, or so I heard.

This Function was **fixed in Tribunal**. You can now force Actors to equip armor, weapons and clothing. So you now CAN do all of the above ☺. Praise to Bethesda.

### Note:

The Equip function can make someone equip an item they aren't carrying, and it will add it to their inventory. However, if you do that then any scripts on the items won't be run. So you need to use AddItem first, then Equip, even though Equip does seem to add them. The script will start if you take the item off the person or out of your inventory and drop. But while it is in inventories the script doesn't start (Forum info, ThePal).

In addition, it appears that Equip will only work on the player after at least one AddItem has been used. -(DinkumThinkum)

The count is optional; if it's omitted then one item will be equipped. You can't use this to force equip more than one item if it's not possible to have more than one equipped at a time (e.g. a cuirass) - it doesn't cause problems but it will still equip only one.

On using equip with variables: it didn't throw any errors, but I got some strange results. It certainly didn't reliably equip the given number.

### Note:

If you equip a potion on a NPC, the NPC will not drink it. You could get around this by making a spell with the same effects as the potion, then removing the potion and adding the spell.

**Sample Script:** This script (Tribunal required) curses an item (a Chitin Club) so that it can't be unequipped anymore. Not even using quick-keys. Bugger! Now you have to fight with a chitin club to the end of your days, which will probably come soon ☺ The player can still use magic however.

```
Begin cursed_item

short state
short OnPCEquip

if ( OnPCEquip == 0 ) ; item is not equipped
    if ( state == 0 ); if club has never been equipped, don't do anything yet.
        return
    else
        Player->Equip, "cursed_club" ; reequip the item!
        MessageBox "The item is cursed, it doesn't leave your hand" ;taunt the player
    endif
```

```

else
    if ( state == 0 ) ;first time equipped. The trap snaps shut
        set state to 1
    endif
endif
End

```

## Detecting if an item has been equipped

**[no fix] OnPCEquip** (is local short variable)

```

Short OnPCEquip
If ( OnPCEquip == 1 )

```

The PC has the object equipped (remains true while object is equipped)  
 This game variable (**needs to be declared!**) gets set to 1 if the player is equipping the calling object. It will remain "true" while the item is still equipped, but gets reset to 0 if the item is unequipped. So, in some cases you might want to manually reset it:

```

if ( OnPCEquip == 1 )      ; when the item is equipped
    [do something]
    set OnPCEquip to 0      ; do this once per equip event...
endif

```

The next time the item is unequipped and equipped again, the functions in [do something] will be performed again. You can also use a status variable to control when an effect will be executed. Note that this can also be processed while in Menu mode:

```

If ( MenuMode == 1 )
    if ( OnPCEquip == 1 )      ; when the item is equipped
        [do something]
        set OnPCEquip to 0      ; do this once per equip event...
    endif
endif

```

This script will execute while you are in the menu, as soon as the item is equipped, while the following will be executed only after you leave the menu:

```

If ( MenuMode == 1 )
    Return
Endif

if ( OnPCEquip == 1 )      ; when the item is equipped
    [do something]
    set OnPCEquip to 0      ; do this once per equip event...
endif

```

An additional **Sample Script** can be found above with the Equip function.

**Notes:** OnPCEquip was successfully tested with the following item types:

- Clothing
- Armor
- Weapons
- Books/Scrolls (see tips and tricks for correct use, though)
- Miscellaneous items
- Usable lights
- Probes

Potions and Ingredients will only register with OnPCEquip when you use SkipEquip at the same time, otherwise the item is apparently "destroyed" before the function registers! Repair objects suffer from this too, and weirder still, apparatus (or alembics anyway) work the reverse way. They only seem to trigger OnPCEquip when PCSkipEquip is NOT set (Forum info / ManaUser).

Apparently books (and maybe some of the other item types that behave strangely?) set SkipEquip to 1 instead of OnPcEquip! See the tips and tricks section on this issue.

## Disabling ability to equip an item

[no fix] PCSkipEquip (is short variable)

```
Short PCSkipEquip  
Set PcSkipEquip to 1
```

Set this to 1 to skip equipping object. Good for popping up messages for breaking seals on books and such. For an extended example look at the SealedTreasuryReport script in the editor. Its also great for using e.g. clothing objects as script triggers in connection with the OnPCEquip function (see my climbing mod for an example, the climbing gear is actually a belt, but of course cannot be equipped).

**Note:** Apparently equipping a book in inventory sets this to one (instead of setting OnPCEquip to one, as it should). See tips and tricks section.

There is a bug associated with using this, that leads to duplication of the item that has the SkipEquip function. I have seen this happen both with QuickKey usage and when normally equipping items from the inventory when OnPCEquip is also called. To bypass this, reset the inventory by adding and removing a dummy item (from within the section of the script that checks the OnPCEquip function). Do not remove the item with the script itself, as this causes a crash (see RemoveItem function). If you have a lot of SkipEquip items, make the item call a global script (StartScript) that adds and removes the item, e.g.:

```
Begin doubling_fix  
  
Player->Additem "Item ID", 1  
Player->RemoveItem "Item ID", 1  
  
StopScript doubling_fix  
  
End
```

**Sample Script:** Here is a short script I made for a werewolf mod, it makes an item non-equipable under certain conditions:

```
Begin non_equipable  
  
; keeps lycanthropic PC's from equipping werewolf hunter items for balancing reasons  
; if the PC equips these before becoming a werewolf, he can wear them until he takes them off  
; but then can't reequip them. So after the first transform he can't equip them again  
  
short PCSkipEquip  
short OnPCEquip  
  
if ( PCWerewolf != 1 ); if player is not a ww, he can use the armor  
    set PCSkipEquip to 0  
    return  
else  
    set PCSkipEquip to 1  
endif  
  
if ( OnPCEquip == 1 )  
    MessageBox "This item is enchanted with werewolf bane-spells. You can not wear it!"  
    set OnPCEquip to 0  
endif  
  
End
```

## Checking for presence of items in inventory

**GetItemCount, "ObjectID" (returns short)**

```
Short objectcount
Set objectcount to ( "Mob_ID"->GetItemCount, "Object_ID" )

If ( GetItemCount, "Object_ID" >= 1 )
```

This function checks the inventory of the calling object and returns the number of objects of type "Object\_ID" it owns.

## Repairing objects

**[no fix] OnPCRepair (is short variable)**

```
Short OnPCRepair
If ( OnPCRepair == 1 )
```

A game variable that gets set to 1 when the PC repairs the object with the script. Requires manual reset.

**RepairedOnMe, "Object ID" (returns Boolean/short)**

```
if ( "daedric_mace"->RepairedOnMe, "repair_journeyman_01" == 1 )
```

This function returns 1 if the calling item is repaired by an item of type "Object ID". Object ID has to be of type "Repair Item" and the calling object must be either weapon or Armor.

## OnRepair

The similar function OnRepair is apparently **broken**. It should get set to 1 when any repair is attempted at the object: "returns true if calling object is repaired at all".

## Worn / equipped object information



**GetWeaponType (returns short)**

```
If ( Player->GetWeaponType == 0 )
;Player uses a short blade
```

**GetArmorType, armorPart\_enum (returns short, -1 to 2)**

```
If ( Player->GetArmorType, 0 == 2 )
;Player wears a heavy helmet
```

These functions are called on an Actor to gather information regarding what the Actor has equipped. GetWeaponType returns the weapon type (see Table 1.1) of the Actor's current weapon. GetArmorType returns the armor weight (see Table 1.3) of the Actor's currently equipped armor part. The armor parts are coded by the numbers listed below (see Table 1.2). HasItemEquipped returns 1 if the Actor has the given item currently equipped and 0 if it does not.

Note: If an NPC has a bow but no arrows and fights using hand-to-hand (no other weapons), GetWeaponType 9 will still return true and HasItemEquipped will return true for the bow.

**Weapon types (Table 1.1):**

| Weapon Type Name         | Type Number |
|--------------------------|-------------|
| Unarmed                  | -1          |
| Short blade, 1 hand      | 0           |
| Long blade, 1 hand       | 1           |
| Long blade, 2 hand close | 2           |
| Blunt, 1 hand            | 3           |
| Blunt, 2 hand close      | 4           |
| Blunt, 2 hand wide       | 5           |
| Spear, 2 hand wide       | 6           |
| Axe, 1 hand              | 7           |
| Axe, 2 close             | 8           |
| Bow                      | 9           |
| Crossbow                 | 10          |
| Thrown weapon            | 11          |
| Arrow<?>                 | 12          |
| Bolt<?>                  | 13          |

**Armor parts (Table 1.2):**

| Armor Part Name | Part Number |
|-----------------|-------------|
| Helmet          | 0           |
| Cuirass         | 1           |
| Left Pauldron   | 2           |
| Right Pauldron  | 3           |
| Greaves         | 4           |
| Boots           | 5           |
| Left Gauntlet   | 6           |
| Right Gauntlet  | 7           |
| Shield          | 8           |
| Left Bracer     | 9           |
| Right Bracer    | 10          |
|                 |             |

**Armor types / weight (Table 1.3):**

| Armor Type Name | Type Number |
|-----------------|-------------|
| Unarmored       | -1          |
| Light Armor     | 0           |
| Medium Armor    | 1           |
| Heavy Armor     | 2           |

### HasItemEquipped "item\_ID" (returns short)

```
If ( Player->HasItemEquipped "chitin club" == 1 )  
    ;you poor dolt!
```

### Sample Script:

When this script is placed on an object, Activating a reference to that object does “damage” to it based on the PC’s current weapon and strength. If the weapon is the specific weapon “Rock Splitter” the object is fully damaged in one hit. When the object is fully damaged, it explodes sending shards into the PC’s face unless he has a shield or a helmet equipped.

```
Begin breakme  
  
float hitsleft  
float hitpercent  
short damage  
short tempdamage  
short weapon  
short doOnce  
short shieldType  
short hasHammer  
short hitRock  
  
if ( doOnce == 0 )  
    set hitsleft to 10000  
    set doOnce to 1  
endif  
  
if ( OnActivate )  
    set hasHammer to ( player->HasItemEquipped "RockSplitter" )  
    if ( hasHammer == 1 )  
        MessageBox "Rock Splitter unleashes its mighty force..."  
        set hitsLeft to 0  
    else  
        MessageBox "You hit the rock with your current weapon..."  
        set weapon to ( player->GetWeaponType )  
        set damage to ( player->getstrength )  
        set tempdamage to 5  
  
        if ( weapon == -1 )  
            set tempdamage to 1  
        endif  
        if ( weapon >= 9 )  
            set tempdamage to 2  
        endif  
        if ( weapon == 4 )  
            set tempdamage to 10  
        endif  
        if ( weapon == 8 )  
            set tempdamage to 8  
        endif  
  
        set damage to damage * tempdamage  
  
        set hitsleft to hitsleft - damage  
    endif  
  
    if ( hitsleft <= 0 )  
        disable  
        set shieldType to ( player->GetArmorType 8 )  
        if ( shieldType == -1 )  
            set shieldType to ( player->GetArmorType 0 )  
            if ( shieldType == -1 )  
                MessageBox "...and the rock shatters sending jagged shards into your eyes."  
                Player->ModHealth -50  
            else  
                MessageBox "...and the rock shatters, deadly shards glancing off your helmet."  
            Endif  
        else  
            MessageBox "...and the rock shatters, deadly shards glancing off your shield."  
        Endif  
    else  
        set hitpercent to hitsleft / 100  
        set hitpercent to 100 - hitpercent
```

```
        MessageBox "...and the rock is %.2f percent damaged but remains intact.", hitpercent
    endif
endif
```

## UsedOnMe function

**UsedOnMe, "Object ID" (returns Boolean/short)**

```
if ( UsedOnMe, Misc_pot_redware_01 )
```

According to helpfile:

"Returns true if the "Object ID" has been used on the calling object. This is used for scripts that make objects do certain things of the player uses an object on it."

According to current knowledge this function is **broken**.

## Moving and placing objects

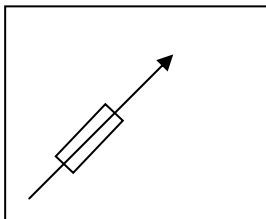
The following functions do not work on the player, NPCs or monsters. They do work on static objects, activators, containers, miscellaneous objects etc. **Note:** Actors, including the PC tend to fall through moving objects after a while. This can be avoided by quickly disabling and enabling a moving object every frame – doing this apparently updates the collision information. Actors can also be equipped with a levitate or slowfall ability to further decrease the chance of falling through objects (for more details see the tips and tricks section on rideable objects by MadMax).

## Moving along an objects axis

**Move axis(x/y/z), units/sec\_enum**

```
Move x, 100
Object_ID->Move Z, 30
```

Moves the object along the selected axis (x, y, or z) at the speed selected. This speed is in units per second (21.3 units per foot). Thus, the distance moved per frame will depend on your frames per second, while the distance moved in a unit time will not. This movement is based on the object's local coordinate system. Thus, a positive y movement will always move the object along its local forward vector:



**Note:** Move will not work on Actors, including the player. However, it will work on dead actors (Forum info / Argent). As with all functions using a "fix", move requires that the object is placed into the game world in the editor, before you can use it in a script:

```
PlaceAtPC "My_Object", 1,1,1
My_Object->Move x, 10
```

will not work if "My Object" has not already been placed, but you can have a local script running on "MyObject" that just uses

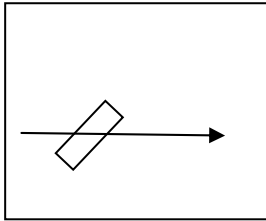
```
Move x, 10
```

## Moving along the world axis

**MoveWorld axis(x/y/z), units/sec\_enum**

```
MoveWorld z, 100  
Object_ID->MoveWorld Z, 30
```

Moves the object along the selected world axis (x, y, or z) at the speed selected. This speed is in units per second (21.3 units per foot). This movement is based on the world axis, thus a positive z movement will always move the object up, regardless of its local rotation: In world coordinates Z is always up / down (increasing upwards), X is east / west (increasing to east) and Y is north / south (increasing to north).



**Note:** MoveWorld will not work on Actors, including the player. Use SetPos for actors.

This is an **example** after a script I once picked up on the forums that makes a platform slowly move out and back once the player stands on it:

```
Begin platform_script  
  
Short PlatformMoving  
Short ActivateMe  
Float Timer  
  
If ( GetStandingPC == 1 )  
    Set ActivateMe to 1  
Endif  
  
If ( ActivateMe == 1 )  
    If ( PlatformMoving == 0 )  
        Set Timer to Timer + GetSecondsPassed  
        If ( Timer <= 15 )  
            "floating_platform_01"->MoveWorld X 10  
        Else  
            Set Timer to 0  
            Set PlatformMoving to -1  
        Endif  
    Endif  
    If ( PlatformMoving == -1 )  
        Set Timer to Timer + GetSecondsPassed  
        If ( Timer <= 15 )  
            "floating_platform_01"->MoveWorld X -10  
        Else  
            Set Timer to 0  
            Set PlatformMoving to 0  
            Set ActivateMe to -1  
        Endif  
    Endif  
Else  
    "floating_platform_01"->SetAtStart  
Endif  
  
End platform_script
```

## CellUpdate

### CellUpdate

**Broken!** According to Bethesda: Updates the current object's cell position. This should be called when moving objects over large distances. The game keeps tracks of objects based on what cell they are in, and if an object moves a cell over from its starting position, it may not get processed correctly when running its script.

The part about not processing correctly is certainly right. Objects can disappear or "warp" if moved too far from the place they were created in. Unfortunately my attempts to use this function always resulted in a runtime error: "need to add function code for function CellUpdate".

**Note:** a way around this problem (requires Tribunal) is to disable and delete (`SetDelete`) the object on a regular basis (for rideable objects upon entering a new cell) and immediately placing a new version (`PlaceItem`) at the very same position using a global script (seen in MadMax boat script from the Fishing Academy Mod). See the Tips and Tricks section for an in-depth explanation by MadMax himself. This works like a charm, because this way the object never really leaves the cell it was created in.

## Setting position (the other way to do movement)

### SetPos, axis, float\_enum\_pos (float\_var with Tribunal/Bloodmoon)

```
SetPos, z, 477  
Object_ID->SetPos X, 466
```

This function (unlike the move and moveworld functions) also works with Actors, including the player. Axis is x, y, or z. The float value sets the position of the calling object to that value. This always refers to the local coordinate system the object is currently in.

**Note:** With Tribunal, this function accepts local float variables, but only within the currently active cells. This is relevant for exteriors, you can not move objects an arbitrary distance, the target location must be within the active cells (current cell of player plus surrounding cells). (Forum info / reposted by Srikandi). Also note that while objects can be `SetPos` to any position (without collision being detected), Actors will still check for collision, and may not move as expected in case of collision (which can be used to good effect for collision detection).

**SampleScript:** This script is made for floating crates in the Mournhold sewer (Tribunal). It demonstrates how `SetPos` and `SetAngle` can be used instead of `MoveWorld` and `Rotate` to produce fluid movements:

```
begin floatAboveStartHeight  
  
float timer  
float swingTime  
float startAngle  
float startHeight  
float currange  
float xvalue  
float zvalue  
float zoffset  
float tmpoffset  
float weightoffset  
float waterlevel  
  
short reset  
short initialized  
  
if ( initialized == 0 ); this section stores the starting height and facing of the object  
    set startAngle to GetAngle, X
```

```

        set startHeight to GetPos, Z
        set swingTime to 1
        set initialized to 1
endif
if ( MenuMode == 0 )
    set waterlevel to GetWaterLevel
    if ( waterlevel > startHeight )
        if ( timer == 0 )
            if ( reset == 0 )
                set timer to Random 100
                set timer to timer / 4
            endif
        endif

        set timer to ( timer + GetSecondsPassed )
        set currangle to GetAngle X
        ;These set the amount to move or rotate depending on framerate:
        set xvalue to 10 * GetSecondsPassed
        set zvalue to 5 * GetSecondsPassed
        ; the crate sways around its x axis:
        ;rotate up
        if ( timer < swingTime )
            set currangle to currangle + xvalue
            SetAngle X currangle
            set zoffset to zoffset + zvalue
        ;rotate down
        elseif ( timer < (swingTime * 3) )
            set currangle to currangle - xvalue
            SetAngle X currangle
            set zoffset to zoffset - zvalue
        ;up again
        elseif (timer < (swingTime * 4) )
            set currangle to currangle + xvalue
            SetAngle X currangle
            set zoffset to zoffset + zvalue
        ;reset timer to zero
        else
            set timer to 0
            set reset to 1
            set zoffset to 0
            SetAngle, x, startangle
        endif

        set tmpoffset to waterlevel
        set tmpoffset to tmpoffset + zoffset
        ; The crate bobs up and down
        SetPos Z tmpoffset
    Else ; Waterlevel is normal
        SetAngle, X, startAngle
        SetPos Z startHeight
    endif
endif
end

```

## Positioning an object in the world or in an interior cell

**Position, float\_enum\_x, float\_enum\_y, float\_enum\_z, float\_enum\_zRot**  
(for outdoors)(floats accepted with expansions)

**PositionCell, float\_enum\_x, float\_enum\_y, float\_enum\_z, float\_enum\_zRot, "cellID"**  
(for interior / exterior cells) (floats accepted with expansions)

```
position -23515, -15355, 3355, 90
Player->position -23515, -15355, 3355, 90
"Actor_ID"->PositionCell, -254, 475, -376, 360, "Balmora, Council Club"
```

The classic application for this function is the teleport ring, transporting the player to certain locations. However, it can also be used to warp NPCs or objects to a new location. Note that in original Morrowind this function only accepted literal values as arguments. (This probably changed with Tribunal: not sure if in all versions or only with the expansions, but: Position/PositionCell can take float variables, but they must be LOCAL variables! (info by Indigo Rage).

**Z\_Rot is not set in degrees (0-360°) but in minutes (1° = 60 min): So, if you want the person to face east, use 5400. South, 10800. West 16200.**

**However, for some reason degrees are used for the player (thanks to Axel for pointing this out).**

### Notes:

Using PositionCell in dialog results isn't reliable, and may cause crashes: some users have no problems with it, but for others it will crash the game, and for some users it may simply fail unpredictably (this doesn't seem to be version-dependent, not sure what causes it). The way Bethesda does this correctly is to use StartScript to start a script that does the teleporting. (Forum Info/Emma).

You should not use this function on items that are in the players inventory, this causes MW to crash (Forum info/Nigedo).

If you are repositioning the player immediately on detecting PC cell or CellChanged (e.g. preventing teleport to a forbidden destination), you may need to introduce a delay of one or two frames before using PositionCell to move the player back out of the cell, otherwise the player may not end up at the correct coordinates. (Forum info / Monica21)

**Sample Script:** A simple teleport ring could look like this:

```
Begin TeleportScript
;Attached e.g. to a ring

short status
short button
short OnPCEquip

if ( MenuMode == 1 )
    return
endif

if ( OnPCEquip == 1 )
    Set Status to 10
    Set OnPCEquip to 0
Endif

If ( status == 10 ); Display menu
    MessageBox "Teleport me to", "Balmora", "Vivec", "Cancel"
    Set Status to 20
```

```

Elseif ( status == 20 ); wait for response
  Set button to GetButtonPressed
  If ( button == -1 ) ; no answer yet
    Return
  Elseif ( button == 0 ); selected Balmora
    Player->PositionCell -21278, -17613, 534, 0, "Balmora (-3, -3)"
  Elseif ( button == 1 ); Selected Vivec
    Player->Position 29872, -82108, 578, 180
  Elseif ( button >= 2 ); selected cancel
    Set status to 0
  Endif
Endif
End

```

Note that both targets are outdoor cells and the different formats used. If you try to teleport to an unsafe place (clipping with an object or out in the void), you will instead be placed at the next safe location.

Sometimes the PositionCell format given above won't work (even though it should): If there are no errors but nothing happens, try replacing, e.g.

```

Player->PositionCell -21278, -17613, 534, 0, "Balmora (-3, -3)"

```

with

```

Player->PositionCell -21278, -17613, 534, 0, "Balmora"

```

I've found that sometimes one method works and sometimes the other.

## Resetting an object to its original position

### SetAtStart

```

SetAtStart
Object_ID->SetAtStart

```

This resets the object to the original position it was given in the editor, before any movement or rotation occurred. For an example, see the moving platform script example under the topic "moving along the world axis". See the *Move* function for a sample script.

## Placing an item near the PC

### [no fix] PlaceAtPC, "Object\_ID", count\_enum, distance\_enum, direction\_enum

```

PlaceAtPC, "Secret Message", 0, 30, 1
PlaceAtPC, "ancestor_ghost", 1, 256, 1

```

This function places a new reference of "Object\_ID" near the player. The function lets you define a direction relative to the player where the object is going to appear and a distance (in units). If that location is not safe (in the air, in a wall, etc), the object will be placed at one of the other axis or at the player's exact location (feet). (Erratum: Thanks to Isildur and Esteban for pointing out that number\_enum and distance\_enum were switched around in previous versions.)

Direction is:

- 0 = front
- 1 = back
- 2 = left
- 3 = right

Note (DinkumThinkum):

*According to Bethesda, on the PC version an overflow loot bag will be created if there are more than 1024 objects in a single cell.*

*However, if you try to put a large number of objects into a cell using 'PlaceAtPC', the game will lockup completely. Don't know the minimum number that will cause the crash, but 'PlaceAtPC p\_restore\_health\_e 2000 0 0' will definitely do it.☺*

*In other words: the game engine won't create an overflow loot bag for excess items that are added to a cell by a script; instead, it just crashes when the maximum number of items per cell is exceeded. I ran into this with 'PlaceAtPC', but I would guess that any script function that adds items to a cell would have the same problem. I would also guess that the crash occurs as soon as the number of items in the cell reaches the number that would normally trigger the creation of an overflow loot bag.*

## Place items near an object



**PlaceAtMe "Item\_ID" count\_enum, distance\_enum, direction\_enum**

Object->PlaceAtMe 'Item\_ID' count\_enum distance\_enum direction\_enum

The PlaceAtMe function works the same as PlaceAtPC without it being centered on the PC. Bloodmoon uses this to place attackers in different places depending on the player's distance at the time. This allows the script to make it appear as though there is a large number of opponents that just keep coming, among other things.

```
;THIS POPS IN A HUNTER AT APPROPRIATE SPOT, INCREMENTS HUNTERCOUNT, AND RESETS TIMER
if ( popA == 1 )
    "active_BM_hunter1"->PlaceAtMe skaal_hunter 1 1 1
    set huntercount to ( huntercount + 1 )
    set timer to 0
elseif ( popB == 1 )
    "active_BM_hunter2"->PlaceAtMe skaal_hunter 1 1 1
    set huntercount to ( huntercount + 1 )
    set timer to 0
elseif ( popC == 1 )
    "active_BM_hunter3"->PlaceAtMe skaal_hunter 1 1 1
    set huntercount to ( huntercount + 1 )
    set timer to 0
endif
```

## Creating new object-references with PlaceItem



**[no fix] PlaceItem "object ID", float\_var\_X, float\_Y, float\_Z, float\_Zrot**

**[no fix] PlaceItemCell "object ID", "cellID", X, Y, Z, Zrot**

These functions are used to create new references to objects. PlaceItem will create a reference to object "Object ID" at coordinate (X, Y, Z) with Z rotation Zrot in the current cell. **Z\_Rot is not set in degrees (0-360°) but in minutes (1° = 60 min): So, if you want the person to face east, use 5400. South, 10800. West 16200.**

The function accepts (local) float variables. PlaceItemCell does the same thing as PlaceItem except it allows you to specify a cell other than the current one in which the object should be created. With either function, if the target cell for the reference is an exterior cell and the given coordinate is outside of that cell, then the reference will be added to the cell containing the coordinate. This is a nice addition that allows you to add things to the world without previously placing them in the editor.

JOG posted the following interesting info on PlaceItemCell:

*Well, When I first tried PlaceitemCell, I thought it's a great function to place items instead of having a "storage-cell" where the objects lie around until the game needs them. I soon realized that you can't refer to those objects by script. (To disable them for example.) The script won't compile until at least one of the objects is in use, and then the disable command would refer to that object, so you still need PositionCell.*

*(Note by GBG: you can circumvent this for many applications by having a script on the placed item, so that you can omit the "fix" and have functions such as disable reference to it by default.)*

DinkumThinkum adds:

*PlaceItemCell would have been perfect for what I wanted to do, until I discovered that the placed NPC disappeared if I saved and reloaded before going to the cell they were placed in.*

*(Note by GBG: This can be avoided in most cases by making the placing of the NPC conditional on the player entering the cell. I still think PlaceItem is a very useful function.)*

### Sample Script:

Putting this script on an object causes it, on activation, to ask the user for an object type and then to create a reference to the specified object at 100 units above the activated reference with 45 degree Z rotation. If the key is selected, it is created at that same coordinate and rotation in the cell "key room" rather than the current cell.

```
Begin makethingsimple

short questionAsked
short button
float myX
float myY
float myZ

if ( MenuMode )
    return
endif

if ( OnActivate == 1 )
    if ( questionAsked == 0 )
        MessageBox, "Create new..." "...Pot" "...Key"
        set questionAsked to 1
        set myX to GetPos X
        set myY to GetPos Y
        set myZ to GetPos Z + 100
    endif
endif

if ( questionAsked != 0 )
    if ( questionAsked == 1 )
        set button to GetButtonPressed
        if ( button == -1 )
            else
                if ( button == 0 )
                    PlaceItem "Misc_pot_redware_01" myX myY myZ 45
                elseif ( button == 1 )
                    PlaceItemCell "misc key" "key room" myX myY myZ 45
                endif
                set questionAsked to 0
                set button to -1
            endif
        endif
    endif
endif
```

```

endif
endif
endif
end

```

## Rotation and angles

Bethesda uses both degrees and minutes (1° degree = 60 minutes) as units in scripting functions:

GetAngle [°] (-180 to 180 °)

Setangle [°]

Position [ min ]

PositionCell [min] *except for the player* [°]

PlaceItem [min]

PlaceItemCell [min]

Rotate [ ° / second ]

RotateWorld [ ° / second ]

## Making an object spin

Similar to the movements described above, you can also rotate objects, around either their local or the world axis and determine the current angle:

**Rotate , axis, angle/sec\_enum**

**RotateWorld, axis, angle/sec\_enum**

```

Rotate, z, -30; rotate counterclockwise, 30° per second, around objects z axis
Object_ID->Rotate, Y, 100

```

Axis can be x, y, or z. Notice that like the move functions the value you are giving Rotate or RotateWorld is a speed setting (not an angle), if you want to turn an object by 90° either use set angle (for an instantaneous change) or use Rotate together with GetAngle to check how far the object has already been turned. These functions can not be used with Actors.

RotateWorld does NOT update an item's position.

If you use RotateWorld in a script, and then call GetAngle, it will always return the same value, regardless of what the angle actually is. Just using rotate does not cause this problem. (Forum info / HeyYou)

## Setting Angles

**SetAngle, axis, float\_enum\_angle**

```

SetAngle, z, 30
Object_ID->Setangle, z, 25

```

This function sets the object to a specific world angle. Axis is x, y, or z. The float value sets the angle (in degrees) of the calling object to that value. This always refers to the local coordinate system the object is currently in.

**Note:** According to console test, this does not affect Actors. With Tribunal, this function accepts local float variables, but only within the currently active cells. This is relevant for exteriors, you can not move objects an arbitrary distance, the target location must be within the active cells (current cell of player plus surrounding cells). (Forum info / reposted by Srikandi). For actors, see the "Face, x, y" function.

Notes (by Simpleton):

*SetAngle* doesn't work how sfd says it does. My guess is nobody ever noticed before because nobody ever uses it with the X or Y axis, which is where it gets funky. I wasn't going to explain how it worked because it's long and complicated and nobody but me cares, but I've got time to kill and there's always a chance somebody will run across this problem and need help.

Probably the oddest thing about *SetAngle*, X/Y is that neither have anything to do with the X or Y axis'. Luckily they do have a system to them that, with a bit of imagination, isn't too hard to understand. The way I found easiest to understand was to imagine a protractor with an arm being held up at a bit of an angle, so that flat side is on the table and the curved part is about 45 degrees off the table. The angle between the table and the protractor is y, and the angle of the arm on the protractor is x.

There are a couple differences, one being that if you set y to 0 the "protractor" is pointing straight up, and a positive x actually goes down instead of up (i.e. the "arm" would go down into the table). So if you set all the angles to 0 and then add 1 to x each frame the object would spin in a vertical circle that is moving down while it's pointing at you. But, if y is 90 (protractor is lying flat on the table) then adding 1 to x each frame would spin the object around the z axis.

Interestingly in the last example if you were to add 1 to \*z\* each frame you would get the same results. That's because while what changing x does depends on the angle in y and vise versa, changing z always does the same thing: spin the object around the z axis.

So, there ya go, and even if you don't understand what I said, you have to give me credit for pulling that picture of a protractor out of my a\*\*.

(ManaUser:)

I'm afraid I don't understand the protractor analogy but the worst thing about *SetAngle* is that it doesn't always save right. Particularly when an object has been rotated on more than one axis, I've noticed it facing some wonky direction after reloading.

**Example script: See SetPos function**

## Scale Functions



**GetScale** (float)  
**SetScale** newScale\_float  
**ModScale** scaleChange\_float

```
If ( doonce == 0 )
    Object_ID->SetScale 0.1
    Set doonce to 1
endif
```

These functions are used to determine or modify the scale of a reference. Any scale you can set must be within (exclusively) 0 and 10 (so you can set it beyond 0.5 and 2, contrary to what was written in the original description by Bethesda) (info by Mode Locrian). The above sample script can thus be used to set scale beyond the limits of what the TES CS itself allows. **Notes:** You shouldn't call "setscale" in every frame, at least not in exteriors or other FPS-critical situations.

The scale will be reset to a value within the 0.5 - 2.0 frame when you reload. So don't use a done-flag to call it only once. Either call it regularly (about all 10 frames) or test for "getscale" and reset the scale when it doesn't fit:

```
if ( GetScale != 5 )
    SetScale, 5
endif
```

Another way is to set it in a Tribunal or Bloodmoon start script. This can be done once, and will make sure the scale is set each time you load a game. (Forum info / JOG).

### Sample script:

A more complex sample script by Bethesda shows how to grow and shrink an object.

When this script is placed on a reference, activating that reference gives the user the ability to change that reference's scale.

```
Begin scalescript
short questionAsked
short button

float direction
float currscale
float tempScale

if ( MenuMode )
    return
endif

if ( OnActivate == 1 )
    if ( questionAsked == 0 )
        MessageBox, "Make this object..." "...Grow" "...Shrink"
        set questionAsked to 1
    endif
endif

if ( questionAsked == 1 )
    set button to GetButtonPressed
    if ( button == -1 )
    else
        if ( button == 0 )
            set direction to 1
        elseif ( button == 1 )
            set direction to -1
        endif
    endif
endif
```

```

        set questionAsked to 0
        set button to 0
    endif
endif
if ( direction != 0 )
    set tempScale to .3 * GetSecondsPassed
    set tempScale to tempScale * direction
    ModScale tempScale
    set currScale to GetScale
    if ( direction == -1 )
        if ( currScale <= .5 )
            set direction to 0
        endif
    else
        if ( currScale >= 2 )
            set direction to 0
        endif
    endif
endif
end scalescript

```

## ***Determining location, relative position and movement***

### **Detecting if player is indoors or outdoors**

**[no fix] GetInterior (returns Boolean/short)**

```
If ( GetInterior == 1 )
```

**Undocumented function!** (Thanks XP-Cagey and Killgore)

This function will return 1 if the current cell is an interior cell and 0 if it is an exterior cell. It will also return 1 if the player is in an interior acting like exterior (GetWindSpeed can be used to detect a fake exterior).

The following is a global sample script by Killgore. If you want to try it out start it by typing "StartScript Outside\_Check" in the console.

```
Begin Outside_Check
short doOnce

if (MenuMode == 1)
Return
EndIf

if (doOnce == 0) ;if you're in some brand new cell
;or it just started

if ( GetInterior == 1 )
MessageBox "1: inside"
elseif ( GetInterior == 0 )
MessageBox "0: outside"
else
MessageBox "mystery else"
endif

set doOnce to 1
Return
endif

if (doOnce == 1)
if (CellChanged == 0) Return
else ;if the player changes cells that frame..
set doOnce to 2 ;it waits an extra frame
endif
Return
endif

if (doOnce == 2) ;then starts over and prints
set doOnce to 0
Return
endif

End Outside_Check
```

### **Determining the players cell**

**[no fix] GetPCCell, "Cell\_ID" (returns Boolean/short)**

```
if ( GetPCCell "Balmora" == 1 )
Set dream to 1
endif
```

The GetPCCell function tests for the player's presence in the specified cell. It returns 1 if the player is in the specified cell, 0 if not. Partial matches are supported, e.g. GetPCCell, "Vivec" will return true for the cells "Vivec", "Vivec, foreign quarter waistworks" and "Vivec, temple", etc.

### Sample Script:

This small Bethesda script checks for the PC leaving a certain area until removing a certain item from an NPC:

```
Begin DrothPost
if ( GetJournalIndex "MS_EstateSale" >= 70 )
    if ( GetPCCell "Mournhold, Geon Auline's House" == 0 )
        "Geon Auline"->RemoveItem "silver dagger_droth_unique" 1
        Journal MS_EstateSale 80
        StopScript DrothPost
    endif
endif
End DrothPost
```

### Distance of one object to another

**GetDistance, "ObjectID" (returns float)**

"ObjectID1"->GetDistance, "ObjectID2"

This function returns the distance (in units) of one object to another. In syntax one, that is the distance between the calling object (to which script is attached to) and the named object. This can be used to trigger attacks or other events, or simply to roughly determine the player's whereabouts for use in a script.

Here is a snippet from an original Morrowind script:

```
; From a script attached to an NPC Ashamanu:
; Ashamanu will give journal entry 60 when player is near
if ( GetDisabled != 1 )
    if ( GetDistance Player <= 256 )
        if ( GetDistance "guar_white_unique" <= 256 )
            if ( GetJournalIndex "MS_WhiteGuar" <= 50 )
                Journal "MS_WhiteGuar" 60
            endif
        endif
    endif
endif
endif
```

### Limitations:

- GetDistance requires that the object given as parameter is placed in the gameworld (in the editor) and has references persist checked (or is naturally persistent such as an NPC).
- Note that you should use this function only with unique ID's or in environments where you exactly know that there is only one instance of the ID – otherwise the Game engine will just grab the first instance of the ID it finds and report that distance – most likely not the distance to the object you want. Thus, a script that warns the player of the presence of a slaughterfish that is closer than 800 units would have to be attached to the ID of the slaughterfish, and check the distance to "player" (which is unique), not vice versa.
- If you determine distance to an object you are moving with Move or MoveWorld, GetDistance will still report the **distance to the original location** of the object (the one you set in the editor). Use GetPos and good ol' Pythagoras ( $c^2 = a^2 + b^2$ ) to determine distances in these instances.
- GetDistance won't work on disabled objects.

### Determining an objects position and facing

**GetPos, axis(x/y/z)**

```
Object_ID->GetPos, z
```

When you are moving objects with the Move/MoveWorld functions described above, you might want to obtain information on its current whereabouts. This functions work on Actors and objects. In the following sample script I used this function to control the movement of a light source (a fire) to make a fire pit where the flames actually start slowly and die down in the evening on a daily schedule – the fire objects original Z position is 511:

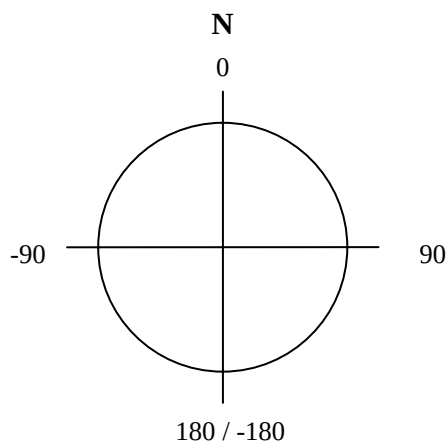
```
Begin _HB_Scheduled fire
short control_fire
; The script is attached to an NPC that guards the fire.
;***** this controls the fires schedule:
if ( GetDistance, "HB_Furn_De_Firepit_camp" < 600 )
    If ( GameHour < 17 )
        if ( HB_Light_Fire_camp->GetPos Z >= 400 )
            HB_Light_Fire_camp->MoveWorld z, -0.1 ; Move fire down
        else
            HB_Light_Fire_camp->disable
        endif
    elseif ( GameHour >= 17)
        HB_Light_Fire_camp->enable
        if ( HB_Light_Fire_camp->GetPos Z < 511 )
            HB_Light_Fire_camp->MoveWorld z, 0.1 ; move fire up
        else
            HB_Light_Fire_camp->enable
        endif
    endif
endif
end
```

### GetAngle , axis(x/y/z) (returns float)

```
If ( Object_Id->GetAngle, z == 180 )
```

The GetAngle function returns the world angle, not the local angle. World angles are from 0 to 180 and 0 to -180 (see figure for z axis).

**Note:** This works on actors and objects, however for the player (and I assume other actors) only the z axis is relevant - GetAngle, x or y always return 0.



### Line of Sight

#### GetLOS, ObjectID (returns Boolean/short)

```
Actor_ID->GetLOS, Player
```

**Undocumented:**  
**GetLineOfSight** (returns Boolean/short)

These functions determines whether the calling object has line-of-sight to the referenced object. It does not seem to work for non-Actor type objects, as far as I could determine. It does not take facing into account, so don't take the "sight" part too literally. (See "Is she looking at me?" in the Tips and Tricks section.)

**Note:** GetLOS is a very slow function, don't let the script call it every frame.

**Sample Script:**

```
Begin balynScript
float timer
short doOnce
[...]; references to journal settings
Set timer to ( timer + GetSecondsPassed )
if ( timer < 5 ); A timer to avoid testing to often (avoids performance problems)
    Return
endif
Set timer to 0
if ( doOnce == 0 )
    if ( GetDistance Player <= 1024 )
        if ( player->GetDistance "hlaalu_loaddoor_ 02_balyn" <= 256 )
            if ( GetLOS Player == 1 )
                ForceGreeting
                Journal DA_Mephala 55
                set doOnce to -1
            endif
        endif
    endif
endif
End
```

**Note:**  
getLOS and getLineOfSight suffer from the same problem with generic NPCs as getDetected (see below)

## ***Determine whether an Actor is detected by another Actor***

**[no fix] GetDetected, "Actor ID" (returns Boolean/short)**

```
If ( GetDetected, Player == 1 )
```

Returns true if **any** calling Actor can detect "Actor ID" (thanks for the correction, ThePal!). This function will return 0 if the Actor is hidden in some form, e.g. is sneaking successfully, or has an invisibility or chameleon spell active. According to the helpfile this is a slow function, do not call it a lot (e.g. make a counter to only call it every 3 seconds).

**Sample script:** The player must approach an object undetected – if not he is "caught"

```
Begin jeanneScript
float timer
short nolore

if ( GetJournalIndex "EB_Bone" < 20 )
    Return
endif

if ( GetJournalIndex EB_Bone >= 40 )
    Return
endif

Set timer to ( timer + GetSecondsPassed )

if ( timer < 5 ) ;this makes sure GetDetected is only called every 5 seconds
    Return
endif

Set timer to 0

if ( GetDistance Player <= 1024 )
    if ( player->GetDistance "com_chest_02 " <=128 )
        if ( GetDetected Player == 1 )
            ForceGreeting ;The player has been caught and will be punished
            Journal EB_Bone 50
        endif
    endif
endif

End jeanneScript
```

GetDetected calling an NPC's ID if there's more than one reference in the game will return nothing. For example:

```
getDetected, "fargoth"
```

will return 1, but:

```
GetDetected "Imperial Guard"
```

Returns nothing. (Not 0, it doesn't return a value). However if you reference an exact NPC:

```
GetDetected "Imperial Guard00000002"
```

A value is returned.

(Forum info / Fliggerty )

## **Determining when the PC leaves a cell**

**[no fix] CellChanged**

```
If ( CellChanged == 1 )
```

CellChanged returns 1 for one frame when player changes cells. It doesn't return 1 for scripted teleporting or magic teleporting. CellChanged returns true almost immediately after the player changes cells. This means a local script running in an interior cell won't fire when the user leaves a cell, but rather when the player enters the cell.

Scripted teleporting may trigger CellChanged if the script is global or targeted (local scripts will not trigger it). (Forum info / Zennorious, Tamandra)

Note:

CellChanged doesn't always trigger, even if the player enters the cell via a normal teleport door. Possibly scripts running in the cell can have some effect on this, somehow.

ForceGreeting seems to muck it up in particular (and no there wasn't a menumode return in that script).

**Sample Script:** In the SlaveScript, which governs freeing slaves in the game, CellChanged is the trigger to disable the slave – the slave has left for a better future:

```
Begin SlaveScript
[...]
if ( slaveStatus == 3 )
    if ( GetCurrentAIPackage == 3 )
        AIWander 512 0 0 0 0 0 0 0 0 0 0
    endif
    if ( GetItemCount Slave_Bracer_Left > 0 )
        Drop Slave_Bracer_Left 1
    endif
    if ( GetItemCount Slave_Bracer_Right > 0 )
        Drop Slave_Bracer_Right 1
    endif
    if ( CellChanged == 1 )
        Disable
    endif
endif
end slaveScript
```

another nice example is the Gateway Haunt's script. This spectre always comes back just when you are not watching:

```
Begin ResurrectHaunt

;town_Sadrith quest
;gateway_haunt resurrects until journal town_Sadrith >= 35

if ( CellChanged == 1 )
    if ( gateway_haunt->GetHealth < 1 )
        gateway_haunt->Resurrect
    endif
endif

end ResurrectHaunt
```

## Detect player traveling



[no fix] GetPCTraveling (returns boolean)

```
if ( GetPCTraveling == 1 )
```

Bloodmoon adds a functions to check whether the PC is traveling (i.e. on a Silt Strider). Will return 1 if traveling/in jail, zero otherwise. This is used in the werewolf change script to stop the PC from changing if either of these are true:

```
if ( PCWerewolf != 1 ) ; DON' RUN IF PLAYER ISN'T WEREWOLF
    return
endif

if ( GetPCinJail == 1 )
    return
endif

if ( GetPCTraveling == 1 )
    return
endif
```

### Triggers for Actors standing on an object

**GetStandingPC** (returns Boolean/short)

returns 1 if PC is standing on it.

**GetStandingActor** (returns Boolean/short)

returns 1 if ANY Actor (including PC) is standing on it.

```
    If ( Object_Id->GetStandingPC == 1)
        [... trigger horrible trap]
    endif
```

These are great functions to trigger events, especially for interior cells. It is also an excellent function to build traps. You can make an "activator" object using the nif files of any static object (hallway, carpets etc.), and trigger certain events once the player (or another Actor) steps on that object.

My sample script is used to light fires in a hall on as soon as the player steps on a particular piece of floor:

```
Begin HBHallLighting
if ( GetStandingPC == 1 )
    set HB_hallfire to 1
endif
end
```

HB\_hallfire is a global variable, used to turn on the fire. This is the script for the fire:

```
Begin HBHallfireon
if ( HB_hallfire == 1 )
    if ( GetPos, z, < -736 )
        MoveWorld, z, 3 ; fire rises, until its reached full height
        if ( GetPos, z, > -780 )
            enable
        endif
    endif
else
    disable
endif
end
```

## Hurting an Actor standing on an object

### HurtStandingActor, float\_HP/s

```
HurtStandingActor -3
Object_ID->HurtStandingActor 1

Float hurt_variable
HurtStandingActor, hurt_variable
```

This function affects the health of an Actor (including the PC) that stands on the object. Positive values will reduce health, negative values heal (hitpoints per second). Using negative values will modify your health even beyond your normal maximum health, so be careful to test this if you want to use this function this way. This function accepts float variables.

**Sample script:** The effect is probably best known from the lava fields:

```
begin lava
if ( menumode == 1 )
    return
endif
if ( CellChanged == 0 )
    if ( GetSoundPlaying "lava layer" == 0 )
        PlayLoopSound3DVP "lava layer", 1.0, 1.0
    endif
endif
HurtStandingActor, 20.0                ;20 pts of damage a sec
end lava
```

## Object Collision Functions



**GetCollidingPC** (returns Boolean/short)

**GetCollidingActor** (returns Boolean/short)

```
if ( GetCollidingPC == 1 )
```

**HurtCollidingActor**, damage\_enum

```
HurtCollidingActor, 100
Object_ID->HurtCollidingActor, 100
```

These functions go on an object to allow it to interact with Actors colliding with it. GetCollidingPC returns 1 if the PC is currently colliding with the object and 0 otherwise. GetCollidingActor works the same as the PC version of the function but will return 1 if any Actor (other than the PC) is colliding with the object. HurtCollidingActor lowers the colliding Actor's health Damage points every second.

**Note:** An object has to have collision to trigger these functions, so if you can walk through the object, none of these functions will trigger.

**Sample script:**

When this script is placed on an object, any Actor touching that object will take damage. A different message is given depending on whether the Actor is the PC or someone else.

```
Begin hurtActor
if ( GetCollidingPC == 1 )
    MessageBox "You scream in pain as you touch the rock."
Elseif ( GetCollidingActor == 1 )
```

```
        MessageBox "Nearby someone screams in pain."
Endif

HurtCollidingActor 100

End
```

## ***Checking activation of an item and activating it***

An item is usually activated when you press the spacebar to "use" it. Most items have a standard action that is performed when they are activated: Doors open, chests display their content, Actors initiate dialogue etc. The OnActivate function allows you to intercept the standard action and do something else or check a condition first:

### **OnActivate**

```
    If ( OnActivate == 1 )
```

This get set to one for one frame when the object is activated. OnActivate resets itself as soon as the function is called, so only one script can report OnActivate successfully, and you should only have one OnActivate call in your script at each moment. OnActivate short-circuits the normal behavior of the object -to perform the standard action of the object once OnActivate has been called, you have to use the activate command:

### **Activate**

```
        activate (if script is on the object to activate)
```

Activate can be used to trigger the standard action of an object, after OnActivate has been called.

### **Standard actions called by Activate are:**

Doors -> Open (Possibly broken in Original MW, with v1.6.1820 it definitely works).  
LoadDoors -> Open (If used with OnActivate)  
Containers -> Open (show contents)  
Books/scrolls -> Display text to read.  
Activators -> Nothing  
Actors -> Dialogue  
Weapon -> Pick up  
Armor -> Pick up  
Miscellaneous -> Pick up

Not tested, but I assume:

Equip. Lights -> Pick up  
Other Lights -> Nothing  
Probes -> Pick up  
Apparatus -> Pick up

### **Example script:**

The following script demonstrates the use of OnActivate nicely. It is attached to a container object, a chest that does NOT advertise its trap like the normal in-game ones do:

```
Begin Trap_script

short done

if ( OnActivate == 1 )
    if ( done == 1 ) ;do-once condition
        Activate
        return
    else
        Cast, "flame", Player ;damage to player
        set done to 1
        Activate ; Call standard action: the chest opens
    endif
endif

End trap_script
```

**Notes:** According to my testing, the activate function can not be used by itself, without OnActivate in the same script. The OnActivate and Activate functions can be in different parts of the script, though. However, Activate will not work before OnActivate has not at least been called once. There have been reports that the object also must have been manually activated at least once within the last 72 hours, so apparently the game eventually forgets that OnActivate has been called previously. Containers require manual activation each load session (forum info / Phaedrus). Play around with this testscript (I had it attached to a door) to see for yourself:

```
Begin GBG_closing_door

Short done

if ( OnActivate == 1 )
    MessageBox "Thank you for activating"
endif

if ( done == 0 )
    If ( GetDistance, Player < 100 )
        set done to 1
        MessageBox "Player close"
    endif
endif

if ( done == 1 )
    If ( GetDistance, Player > 200 )
        MessageBox "Sesame"
        Activate
        set done to 0
    endif
endif

EndIf

End
```

There have also been various reports about difficulties with on OnActivate in general, in my experience, avoiding to use more than one OnActivate in a script is the safest way to go. Also be aware that for items in your inventory, OnPCEquip may be the function to use instead of OnActivate, as OnActivate does not get called by moving the item over the Player icon.

Info from the UESP: There is an undocumented feature in the Activate function by specifying the player after the function, for example:

```
begin RemoteContainer
    short OnPCEquip
```

```
if ( OnPCEquip == 1 )  
    set OnPCEquip to 0  
    "dh_remote_chest_01"->Activate, player  
endif  
end
```

If the container is persistent (references persist) this script should open the container wherever the player is. This is a great way to create 'carryable' containers by attaching a script similar to the above to a ring or similar item.

## ***Locking and Unlocking doors or chests***

**Lock, short\_enum\_locklevel**  
**Unlock**

```
My_Door->Lock, 50
```

**GetLocked (returns Boolean/short)**

```
If ( GetLocked == 1 )  
    Unlock  
Endif
```

(only Door and container objects)

These functions are used to lock and unlock doors or containers. The GetLocked function returns 1 when the calling object is locked. Lock locks the object with the lock level specified (0-100). Lock 0 has an odd effect though. The door/container will be neither openable nor pickable. Unlock removes any lock, regardless of lock level.

### **Example script:**

Here is a sample script by qwert, which makes a chest function as a security skill-training device by constantly relocking it:

```
Begin PC_Security_Skill_Trainer  
  
float timer  
  
if ( menumode == 1 )  
    return  
endif  
  
set timer to timer + GetSecondsPassed  
if ( timer > 10 )  
    set timer to 0  
endif  
  
if ( timer == 0 ) ;using a timer to relock after 10 seconds pass  
    "Storm_Chest_Trainer"->Lock 50  
endif  
  
End
```

## ***Animating objects***

There is a group of functions that allows you to play specific animations that are defined in a model (.nif file). You can find out about the animation group names by loading a model into the preview window and then skipping through the different animation groups or by looking at "base animation" windows in the "Character" menu. An excellent summary of Actor animation groups can be found here:

<http://www.preik.net/morrowind/animationsgroups.html>

but only the ones listed in the base animation window can be called by this function.

Additional animations can be loaded to a model via the animation button in the object window. See the Dancing girls in "Suran, Desele's house of earthly delights" for an example. Not all models have animation groups, but the different banners (under activators) are good examples to see what is meant (see example below). Examples for GroupName are: idle, idle2, idle3, walk, etc.

These functions do not work on the player character.

**PlayGroup, GroupName, [Flags]**

```
PlayGroup, walk, 1
```

Plays the animation group defined by GroupName. Optional flags can be used to start the group in different ways (see below).

#### **LoopGroup, GroupName, Number\_enum, [Flags]**

Plays the animation group defined by GroupName. The animation will be looped the number of times specified, and then return to the Idle animation. Optional flags can be used to start the group in different ways (see below).

#### **Flags:**

*0 = Normal*

The current animation will finish its full cycle, and the new animation will start from its beginning.

*1 = Immediate Start*

The current animation will stop regardless of the frame it is on, and the new animation will start from its beginning.

*2 = Immediate Loop*

The current animation will stop regardless of the frame it is on, and the new animation will start at the beginning of its loop cycle.

**Note:** With Bloodmoon installed (not necessarily Bloodmoon.esm selected) some of NPC animations are crosswired. When called from console, they may look correct, but if you insert NPC->PlayGroup, group, 1 into the script, you may be unpleasantly surprised to see a different animation than you expected (Forum info / Kir). You may have to experiment to find the correct animation (Check out the info in the section on AIWander for a list of NPC idle movements, or download the NPC Animation Explorer mod to see all possible animations ingame: [http://www.angelfire.com/rpg2/mad\\_weather/animexp.htm](http://www.angelfire.com/rpg2/mad_weather/animexp.htm)).

**Note:** NPCs may play a different animation on the upper body while the lower body does the scripted animation. This may also be version-dependent(?), but in any case these functions appear to be unreliable when used with NPCs.

#### **Example Script:**

This original script is attached to all the outside banners and makes them move differently depending on the weather:

```
begin OutsideBanner

;this script is for a banner object outside that
;animates in the wind.
;Idle is still, Idle2 is a little breeze, and Idle3 is a large breeze

short ran

if ( MenuMode == 0 )
    set ran to random 100
    if ( ran < 30 )          ;30% chance the flag does something new
        if (GetCurrentWeather >= 5 )          ;thunder, ash, or blight
            LoopGroup, Idle3, 5
        endif
        ;the last anim called in this script is the one it will play
        if ( ran <= 10 )
            PlayGroup, Idle
        elseif ( GetCurrentWeather < 5 )
            PlayGroup, Idle2
        endif
    endif
endif

end if

end OutsideBanner
```

## SkipAnim

Causes the current animation to not be played for this frame.

## Enabling and disabling objects

### Enable

### Disable

```
"ObjectID"->enable
```

### GetDisabled (returns Boolean/short)

```
If ( GetDisabled == 1 )  
    Return  
Endif
```

The *Disable* function makes an object completely vanish from the game world, meaning it's neither rendered nor processed (attached scripts are still active, however). The *Enable* function makes a disabled object visible and processed again. *GetDisabled* (returns 1 if object is disabled) can be used to obtain the current status of an Object. These functions are very powerful and could e.g. be used to swap different models of statics (normal house replaced by house burnt to the ground, etc). It is e.g. used for the stronghold building process. Note that disable should not be called every frame, as this will cause a drop in framerate. Instead, use GetDisabled or some other condition to ensure the object is disabled only when necessary.

### Example script:

An example from the game is the SlaveScript that makes freed slaves vanish once the player leaves the cell:

```
begin slaveScript  
  
short slaveStatus  
short doOnce  
short NoLore  
  
[other slaves status checks - check original script!]  
  
if ( slaveStatus == 3 )  
    if ( GetCurrentAIPackage == 3 )  
        AIWander 512 0 0 0 0 0 0 0 0 0 0  
    endif  
    if ( GetItemCount Slave_Bracer_Left > 0 )  
        Drop Slave_Bracer_Left 1  
    endif  
    if ( GetItemCount Slave_Bracer_Right > 0 )  
        Drop Slave_Bracer_Right 1  
    endif  
    if ( CellChanged == 1 )  
        Disable ;***** Make slaves vanish once freed and player leaves  
    endif  
endif  
  
end slaveScript
```

### Caution: disabling lights

There seems to be a game engine issue with disabled lights – Actors and some other types of objects still seem to be illuminated, while the world around is not. I have not thoroughly tested if this can be avoided, but a suitable workaround is to physically remove the light to a remote location (e.g. a few meters below the floor) instead of disabling it. Another trick comes from Indigo: If you enable a light that is set to "negative" (which means its generating

darkness instead of light) after you disable the normal light, the illumination problem goes away.

#### **Note:**

When you are in an exterior cell (haven't checked with interiors, but presumably it applies in interiors as well) any object you disable in the loaded cells while you are in that cell will retain its collision properties in the game world until the next time that exterior cell is loaded - i.e. you disable a fence piece through a script where that fence piece is in the same exterior cell as your PC. You still can't walk through the area where the fence piece is disabled until you exit and reenter that exterior cell. Similarly, you subsequently reenables that fence piece that was disabled when your PC entered that exterior cell. You can walk right through that fence piece until the next time that exterior cell is reloaded by the game.

To fix this use the "FixMe" command in your script after you disable or enable a game object that exhibits collision. You should check that you are in the same exterior cell as the game object before you execute the FixMe command - if you aren't you don't need FixMe. This is useful in global scripts that enable or disable objects with references persist so you could be in any cell when the object is enabled or disabled. Using FixMe will cause your PC to also move 128 game units (hardly noticeable in most cases other than as a "stutter") when it executes but since there is no other script command that reloads a cell that is something you will have to live with. Using FixMe will cause the collision to be reset for the cell FixMe reloads so is a way to immediately fix collision on script enabled/disabled objects. (Forum info / [Rougetet](#))

## **Deleting a reference completely**



[no fix?] **SetDelete flag\_enum**

SetDelete 1

The SetDelete function can be used in combination with Disable to remove an object more completely. SetDelete 1 marks a reference for deletion and SetDelete 0 clears that flag. This can be useful in optimization. Certain objects have scripts on them which simulate picking them up by disabling the activated reference and adding a new object to your inventory. This leaves an ever-present, but disabled, reference at that location (which eats up memory and processor time since the script on the disabled reference is still run every frame). If the reference is marked for deletion then it is essentially gone. If the reference came from the master file, it is still there but knows it shouldn't be so has no art and no scripting. If was created in game, it will actually be deleted.

#### **Notes and cautions:**

SetDelete will crash Morrowind if there is any other function operating on the object in the same frame. It will also crash if the object emits sounds (or plays animations with linked sounds) and is disabled and deleted in the same frame (forum info / JOG).

This Script **will crash**:

```
Begin _spell_effect
float timer
rotate y 120 ; crash caused by this
```

```

if ( timer < 3 )
    set timer to ( timer + GetSecondsPassed )
else
    disable
    setdelete 1
endif

```

A solution is to use first disabling the object and then using GetDisabled and Return to safely delete the object:

```

Begin _spell_effect
DontSaveObject
float timer
if ( GetDisabled == 1 )
    setdelete 1
    return
endif
rotate y 120
if ( timer < 3 )
    set timer to ( timer + GetSecondsPassed )
else
    disable
endif

```

Another solution was proposed by Soralis, using a local variable "deletobj" as a flag:

```

if ( deleteobj = 1 ) ;Local variable, set when you want to delete
    if ( deletetimer == 0 )
        Disable
    endif
    if ( deletetimer < 10 )
        set deletetimer to ( deletetimer + 1 )
    endif
    if ( deletetimer == 10 )
        SetDelete, 1
    endif
    Return
endif

```

Always call SetDelete from a local script assigned to the object you want to remove. Never use Object->SetDelete 1 as this usually causes crashes when Morrowind is running. However, starting a targeted script on the object and then using SetDelete in that script works fine.

Don't delete items in inventory: this results in incorrect encumbrance. If necessary, Drop can be used to remove the item from inventory before deletion.

It has been reported that magic effects can cause problems with SetDelete (forum info / Dan\_Wheeler). I (melian) am not sure exactly what this refers to, but I do know that if an NPC is deleted too quickly after another NPC casts a spell on it (scripted cast), the game will crash. If this causes you problems, use a timer to wait before deleting the object (you may need to wait a few seconds).

## ***Don't save changes to an object***

### **DontSaveObject**

Call this function if changes to the object are not to be saved to the savegame.

Use DontSaveObject on objects that:

1. Can be enabled/disabled during the game, such as the stages of building a stronghold.
2. Objects that might be moved, such as rideable objects.

By using DontSaveObject you will avoid that annoying "save game data has changed" error message that will occur when load a saved game and the object's state has changed, ie: it has moved, or has been disabled/enabled. This is due to the fact that object data is stored in the save game data, such as if the object is enabled or disabled, or if it has been moved (like a rideable object) (Forum info / IndigoRage).

In the original game, it's used in the SignRotate script and the following one:

### **Sample Script:**

```
Begin diseaseAscended
DontSaveObject

;ascended sleeper has all the blight diseases for some reason...
if ( CellChanged == 0 )
    return
endif
AddSpell "ash woe blight"
AddSpell "black-heart blight"
AddSpell "chanthrax blight"
AddSpell "ash-chancre"

End
```

# Scripting NPCs: AI and Movement

## *Make an NPC walk to a new location*

**AITravel, float\_enum\_x, float\_enum\_y, float\_enum\_z, [reset]**

```
Actor->AITravel, 1359, 2700, 1045
```

To make an NPC walk between different defined places in the game world you use the AITravel function.

The variables x, y, z are world coordinates. You can determine these by moving your camera to the desired endpoint of the movement or by selecting a path grid point or an object nearby coordinates are displayed below the object window. The usage of the optional reset flag is unknown.

When using this function in scripts it is important to provide conditions where the AI package is called only once. Consider the following NPC bound script

```
Begin Travel
AITravel, 1359, 2700, 1045
End Travel
```

The previous script will not work, as the script “fires” continuously, and the effect is that the NPC freezes and never performs the desired movement.

```
Begin Travel
Short do_once

If ( do_once == 0 )
    AITravel, 1359, 2700, 1045
    Set do_once to 1
endif
End Travel
```

This should work; the NPC will wander to the designated coordinates as soon as his script becomes active, meaning as soon as his cell is loaded.

## Checking whether an NPC has performed his movement

**GetAIPackageDone (returns Boolean/short)**

```
if ( GetAIPackageDone == 1 )
    [do something]
endif
```

To check whether an NPC has arrived at his location you can use the *GetAIPackageDone* function. This function returns 1 for one frame when the current AI package has finished.

Use this to perform a check whether a movement has been finished. The following script is an example of how to link several *AITravel* commands within one script, using a state variable and an *if-elseif* structure:

```
Begin TravelLoop
short state
```

```

float timer

if ( menumode == 1 ) ; if menu is open don't process
    return
endif

;start walking
if ( state == 0 )
    if ( player->GetDistance HB_adros_darani < 5000 )
        set state to 5
    endif

;***** He begins his trip
elseif ( state == 5 )
    SetHello 0
    AITravel -8144, -19409, 728 ;new co-ords point 1
    set state to 10

elseif ( state == 10 )
    if ( GetAIPackageDone == 1 ) ;he's reached point 1
        set state to 40
    endif

elseif ( State == 40 )
    AITravel -9147, -19459, 720 ; new co-ords point 2
    set State to 50

elseif ( state == 50 )
    if ( GetAIPackageDone == 1 ) ;he's reached the point 2
        set state to 60
    endif

elseif ( state == 60 )
    AITravel -8144, -19409, 728 ;new co-ords point 1
    set state to 70

elseif ( state == 70 )
    if ( GetAIPackageDone == 1 ) ;he's reached point 1
        set state to 80
    endif

elseif ( state == 80 )
    AITravel -6640, -18496, 1040 ;new co-ords point 0
    set state to 90

elseif ( state == 90 )
    if ( GetAIPackageDone == 1 ) ;he's reached point 0
        set state to 0
    endif

endif

End TravelLoop

```

Good examples of scripting using *AITravel* are also found in the “lookoutsript” (Fargoth hiding his treasure) and the CharGenWalkNPC script (The guard that walks you through the ship at the beginning of the game). Or look at my "Traveling Merchants" plugin for true *AITravel* madness ☺ !

If you plan on using this function extensively you should be aware of the following problems: When you leave a cell with an Actor that is just performing its *AITravel* command, or if you rest, the script will never detect the *GetAIPackageDone* signal, meaning your NPC gets stuck on its path once you return or finish sleeping. The following simple code can be used to get the script going again (this is for the above script)

```

; ***** Stalled Script rescue - recovers script after leaving a cell or resting
If ( Player->GetDistance, HB_adros_darani < 5000 )
    if ( GetCurrentAIPackage == -1 ) ; check for idleness
        set timeout to ( timeout + GetSecondsPassed )
        if ( timeout >= 3 ) ; wait some time.
            ; Short instances of idleness always occur
            set state to ( state - 10 ) ; stall will occur at
            ; AIPackageDone - jump to "wander" again.
            set timeout to 0
        endif
    else
        set timeout to 0
    endif
endif
endif

```

## ***Make an Actor turn or face a certain direction***

**Face** *x\_enum, y\_enum*

```
"ActorID"->Face, -1334, 334
```

To my knowledge this function does not accept variables (at least under version 1.6.1820), but there have been reports that it did - possibly this is version dependent. This function Makes an NPC face towards a specified x/y coordinate. Apparently interrupts current animations. Using this on wandering NPCs makes them stop, face wherever, then continue on wherever they were wandering to, as soon as the facing movement is over. (Forum info / JOG, Dan\_Wheeler).

## ***Define random Actor movement***

**AIWander, range\_enum, duration\_enum, time\_enum, [idle1], [idle2], [idle3], ...[idle9], [reset]**

```
"Actor_ID"->AIWander, 512, 5, 0, 0, 20, 0, 0, 10, 30, 0, 0, 0
```

This is the random movement algorithm that most of the NPCs in the game are using. The NPC travels along the path grid, changes direction randomly, and performs idle movements in between.

- Range: determines the distance the Actor or creature will roam from his origin.
- Duration: probably the time (in hours) the mob will perform the package (before it is reset, which seems to happen if you leave or sleep, not sure?)
- time: presumably determines the start time for the package if it has a duration
- [idle1], ...[idle9]: chances of idle movements.

The Idles are (as tested in game):

- Human male:
  - Idle1: Stand still
  - Idle2: Shifting weight from one leg to the other
  - Idle3: Looking behind
  - Idle4: Scratching head, shake head
  - Idle5: Shifting clothing or armor on shoulder
  - Idle6: Yawning and stretching
  - Idle7: Looking at fingers and looking around furtively
  - Idle8: Putting hand to chest, as if having heartburn
  - Idle9: Reaching for weapon, then touching head
- Human female - as above but:
  - Idle5: Hand on hip

- Khajiit female - as Human male but  
Idle9: Scratching head, shaking head

To let an Actor stand in one spot you can use: AIWander, 0, 0, 0

**Note:** The number of idles and some of the descriptions were listed wrongly in previous versions (corrected with 8<sup>th</sup> edition - credits go to Whoopa).

Here is an **example script** that displays all idles in series (this may be useful to check out which ones you want your NPC to use):

```
Begin Animtest

float timer
short count
set timer to ( timer + GetSecondsPassed )

if ( timer > 10 )
    set timer to 0
    set count to ( count + 1 )
    if ( count >= 18 )
        set count to 0
    endif
endif

if ( count == 1 )
    AIWander 0, 0, 0, 100, 0, 0, 0, 0, 0, 0, 0, 0
    MessageBox "Idle 1 , 100"
    set count to ( count + 1 )
elseif ( count == 3 )
    AIWander 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 0, 0
    MessageBox "Idle 2 , 100"
    set count to ( count + 1 )
elseif ( count == 5 )
    AIWander 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 0
    MessageBox "Idle 3 , 100"
    set count to ( count + 1 )
elseif ( count == 7 )
    AIWander 0, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0
    MessageBox "Idle 4 , 100"
    set count to ( count + 1 )
elseif ( count == 9 )
    AIWander 0, 0, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0
    MessageBox "Idle 5 , 100"
    set count to ( count + 1 )
elseif ( count == 11 )
    AIWander 0, 0, 0, 0, 0, 0, 0, 0, 100, 0, 0, 0
    MessageBox "Idle 6 , 100"
    set count to ( count + 1 )
elseif ( count == 13 )
    AIWander 0, 0, 0, 0, 0, 0, 0, 0, 0, 100, 0, 0
    MessageBox "Idle 7 , 100"
    set count to ( count + 1 )
elseif ( count == 15 )
    AIWander 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 100, 0
    MessageBox "Idle 8 , 100"
    set count to ( count + 1 )
elseif ( count == 17 )
    AIWander 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 100
    MessageBox "Idle 9 , 100"
    set timer to 0
    set count to ( count + 1 )
endif

End
```

## Making Actors activate objects

**AiActivate "Object ID"**

**AiActivate ObjectID [reset]**

Actor->AIActivate "Object"

In the words of Bethesda: "This package tells the Actor to activate the specified ObjectID. A powerful and admittedly underutilized and undertested package."

In standard Morrowind this function appeared to be mostly **broken**, except for making an NPC quaff a potion. **With Tribunal** it seems to have been fixed, at least to some extent. I could successfully use it to get an NPC to pick up a weapon, open a normal door and go through a load door. LoadDoors only work if the door marker it teleports to is in the same interior cell, or within the loaded (PC's current and surrounding) cells in exterior cells – otherwise the game crashes. I also tested it successfully with an activator (switch to open Ghostgate).

The usual precautions with AI functions apply (make sure the Actor is not too far away, have a good AI grid in place, make sure nothing can get in the way, etc.). Although not thoroughly tested, I didn't seem to get an *AIPackageDone* signal, but other conditions can be constructed (see examples below) to set the NPC to a different AIPackage again.

| Object Type               | Activation                                                                                                                                                                                                             |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NPC                       | Activating NPC will approach and circle the target NPC                                                                                                                                                                 |
| Container                 | Opens                                                                                                                                                                                                                  |
| Door                      | Opens                                                                                                                                                                                                                  |
| Load door                 | Opens/teleports (same cell only)                                                                                                                                                                                       |
| Weapon, armor, misc., etc | Picks up                                                                                                                                                                                                               |
| Book/Scroll               | Reads <meaning what to an NPC?>                                                                                                                                                                                        |
| Activators                | Execute as defined by script. If the activator doesn't have an OnActivate block (or doesn't have a script), the Actor will try to activate it anyway, in much the same way as 'activating' NPCs (approach and circle). |

### Note:

If you place the object to be activate too high, the NPC will still approach and circle underneath it.

**Sample Script:** these are just a few testing scripts I made. They show how you can set conditions to determine when the NPC has finished his action.

```
Begin TT_opendoor

short doonce
short AIState

if ( doonce == 0 )
    if ( GetDistance, Player < 400 )
        AIActivate TT_door
        set doonce to 1
    endif
elseif ( doonce == 1 )
```

```

        set AISTate to GetCurrentAIPackage
        MessageBox "Package = %g", AISTate
        if ( TT_door->GetAngle, z != 180 ); As soon as door starts rotating
            MessageBox "Done"
            AIWander 30, 5, 0, 0, 20, 0, 0, 10, 30, 0, 0
            set doonce to 2
        endif
    endif
end

```

```

Begin TT_pickmace
short doonce
short AISTate
if ( doonce == 0 )
    if ( GetDistance, Player < 400 )
        AIActivate TT_daedric_mace
        set doonce to 1
    endif
elseif ( doonce == 1 )
    set AISTate to GetCurrentAIPackage
    MessageBox "Package = %g", AISTate
    if ( GetItemCount, TT_daedric_mace >= 1 ); when NPC has the mace in his inventory
        MessageBox "Done"
        AIWander 512, 5, 0, 0, 20, 0, 0, 10, 30, 0, 0
        set doonce to 2
    endif
endif
end

```

```

Begin TT_openloaddoor
short doonce
short AISTate

if ( doonce == 0 )
    if ( GetDistance, Player < 400 )
        AIActivate TT_door
        set doonce to 1
    endif
elseif ( doonce == 1 )
    set AISTate to GetCurrentAIPackage
    MessageBox "Package = %g", AISTate
    if ( GetPos, y > 2000 ); position has changed. Loador target is in same cell
        MessageBox "Done"
        AIWander 30, 5, 0, 0, 20, 0, 0, 10, 30, 0, 0
        set doonce to 2
    endif
endif
end

```

## Following and Escorting

**AiFollow**, "Actor ID", duration\_f\_enum, x\_f\_enum, y\_f\_enum, z\_f\_enum, [reset]

**AiFollowCell**, "Actor ID", "Cell ID", duration\_f\_enum, x\_f\_enum, y\_f\_enum, z\_f\_enum, [reset]

Actor->AIFollow, "Mob2ID", 0, 0, 0, 0

The "Follow" AI-package makes an Actor closely follow another. You can use this to make an NPC or creature follow the player, but you can also use it to make NPCs and creatures form a caravan. The following excerpt from one of my own scripts shows the unconditional use of the function:

```

elseif ( state == 20 )
    HB_guar_pack_adros_ -> AIFollow, HB_adros_darani, 0, 0, 0, 0
    AITravel -8144, -19409, 728 ;new coords point 1
    set state to 30

```

Since there is no duration or destination location given, the guar will follow the NPC until another command is given. As with other AI-commands, make sure you set conditions so that each AIFollow command is issued only once, not every frame.

The duration, CellID and x, y, z destination coordinates set conditions that once fulfilled will terminate the AIPackage (which you can test with the GetAIPackageDone function as describe for AITravel above). The AiFollowCell function allows you to set an interior cell as the destination.

The meaning of the optional reset option is currently unknown.

**AIEscort, "Actor ID", duration, x, y, z, [reset]**

**AIEscortCell, "Actor ID", "Cell ID", duration, x, y, z, [reset]**

This function makes an Actor lead another actor or the player to a certain point. The Actor will wait for the follower if the distance becomes too great, and will resume its path when the follower approaches again (Thanks to Kir for this info). You see an **example** of this in character generation: the guard who escorts you up from the ship's hold to the ramp on the second level. (Thanks to MisterSmileyFaceDude). The meaning or use of [reset] is unknown.

## Checking which AI package is currently executed

For complex Actor scripting it might be valuable to know which AI mode is currently active and make script actions dependant on it.

**GetCurrentAIPackage (returns short)**

```

If ( GetCurrentAIPackage == 2 )
    [do something]
endif

```

The returned values are:

|          |    |
|----------|----|
| None     | -1 |
| Wander   | 0  |
| Travel   | 1  |
| Escort   | 2  |
| Follow   | 3  |
| Activate | 4  |
| Pursue   | 5  |

Casey Tucker wrote a few interesting notes about Morrowind's AI system:

*These are a few notes on Morrowind's AI system, as much for myself as for anyone wishing to look them over. The most I noticed the AI is incredibly stubborn. There are quite a few unpredictable script functions that deal with AI. I find that the AIFollow is actually the most stable of them.*

*When working on "The Triumvirate" quest/war mod, if the player went to Dragonhold Castle to join the Legion, I had to have the player's Imperial Centurion, Sensius Tanarii, show the player where his barracks room is.*

*Initially I had Sensius follow the player until you wandered around long enough to find your*

barracks room. Sensius would confirm the location and disappear, once Proximus Taiatius, (your legionary roommate) noticed you.

(GetTarget Player == 1 Usually works - For the desired effect, GetDistance is more accurate.)

So, that was sort of the opposite of I wanted. I wanted Sensius to take the player to their room. When I first actually thought of this, I tried thinking of several ways to get an NPC to "AiEscort" through several cells to the given coordinates. Let me tell you of my findings on AIEscort. It's either broken, works only in exterior cells, or I was doing something wrong. Regardless, I noticed that the NPC did not budge at all from their location, though they WERE set to guard the player. Upon further research, I saw that during the CharGen process, when the prison ship guard appears to be escorting you to the door, Bethesda actually used AIWander, making checks to see if the player was following or not. From this I drew the conclusion that AITravel would be so much more convenient and stable. I only use AIEscort to get a stationary actor to guard the player if they came under attack. Useful for battle scenes when you want it to appear that two forces are fighting each other, without having an entire side follow the player. About mass battles... I will lead into that later.

First, I must explain how to get an NPC to escort a player, from point A: an interior cell, through several interior and exterior cells, to point B: also an interior cell. First things first.. we don't want the NPC to get stuck. You'll need some pathgrids. A few notes about pathgrids: the NPC doesn't seem to care about the gridpoints' Z coordinate [This doesn't appear to be correct. See notes below]. That's a very good thing. Pathgrids are like magnetic roads. The NPC is seemingly "magnetically attracted" to that pathgrid, and their AIWandering, AITraveling, and AIFollowing will follow along those grids. To see how these grids work, go in-game, to Ebonheart or another semi-crowded city, and type in the console: 'TPG'. That will show the pathgrids - you'll see the NPC's walking between the two yellow "rails" that are drawn out, as if they were trains. An NPC will reluctantly pull away from a pathgrid if led so by the player, need to walk around something blocking their way, chasing someone/thing, and such scenarios.

Now that I got pathgridding done, I noticed another thing. AITravel is not "unstable" per se, but not very reliable. Two factors come into play here, that dictate if the NPC will blatantly ignore it's orders or do as told. The first, I noticed, was the AI distance setting, changeable by FPS optimizer or in-game menu. Makes sense. However, the next one is rather annoying. The NPC cannot get distracted or their travel will stop. This means you'll have to set their "Hello" distance to 0 during their little stroll, otherwise they will stop and greet the player instead.

Now we come to the point where the NPC reaches a load door through cells. The game will CTD if the NPC activates it. Here is a snippet of an "escort" script I wrote, that gets the NPC through a load door.

```
Begin Sensius_Escort_Script
Short EscortOnce
Float Timer
;In my theory, a load door is simply a PositionCell function referencing to the PlayerSaveGame
once the door is
;activated. (By the player.) For an NPC to activate a load door would, as this "door script"
seems hard-coded
;strictly to the player, would call an error that would cause a CTD. This script creates an
artificial load door, by
;replacing the "Player" function references to the NPC in question. As this is concerning a
REAL load door, I cannot
```

```

;have the NPC activate it. Rather, the NPC teleports when they are close enough to the door
for activation, and
;a certain variable is set, accompanied by a 3D sound to finalize the effect. NPC load doors
are possible. The only
;problem is that the NPC disappears rather than shows an animation of entering the door - but
it's not much worse
;than in Oblivion, when the NPC just fades.

If ( MenuMode == 1 )
Return
Endif

If ( EscortOnce == 0 )
"Sensius Tanarii"->AiTravel 4990.000 5200.000 15456.000
Set EscortOnce to 1
Endif

If ( EscortOnce == 1 )
If ( "Sensius Tanarii"->GetPos, X <= 5060 ) ;I am "boxing in" the coordinates. When he stands
in this area, the door is activated.
If ( "Sensius Tanarii"->GetPos, X >= 4930 )
If ( "Sensius Tanarii"->GetPos, Y <= 5360 )
If ( "Sensius Tanarii"->GetPos, Y >= 5177 )
"Sensius Tanarii"->PlaySound3D "Door Latched Two Open" ;
"Sensius Tanarii"->PositionCell 3186 5191 15050 0 "Dragonhold Keep - Main Entry Hall"
Set EscortOnce to 2
Endif
Endif
Endif
Endif
Endif
...
```

*That should explain how it would work. Coordinates are often more reliable than a "GetAIPackageDone" function, mainly because that function is easily broken by several factors.*

Some notes on the notes (!)

- It's been reported by others (and I've found myself) that the z-pos of pathgrid nodes **does** matter (at least in interiors): if a node isn't well above the floor the actor can get "stuck" at that point and refuse to move past it.
- AIEscort has been used successfully by a number of modders (and in Bethesda's scripts, e.g. CharGenWalkNPC). That's not to say it's without problems, but it is certainly useable. However, since AIEscort may not be viable in some circumstances, the above notes may help you develop an alternative approach should one be necessary.

## ***Forcing sneak movement***

**ForceSneak**

**ClearForceSneak**

"Actor\_ID"->ForceSneak

**GetForceSneak ( returns Boolean/short)**

If ( "actor\_ID"->GetForceSneak == 1 )

The *ForceSneak* command puts the Actor in sneak mode, all movement will be performed in sneaking. *ClearForceSneak* terminates *ForceSneak* mode. Unfortunately there does not seem to be a corresponding command to force running (added with Tribunal). *GetForceSneak* returns one if *ForceSneak* mode is active on the calling Actor. Check the LookoutScript script for an example. Here's a snippet:

```

elseif ( walkstate == 2 )
    Fargoth->ForceSneak ; enter sneak mode
    Fargoth->AiTravel -11468.595,-71511.531,173.728 ;goes to tree
    set walkstate to 3

elseif ( walkstate == 3 )
    if ( Fargoth->GetAiPackageDone == 1 )
        ;Fargoth->Equip "torch_infinite_time_unique"
        set walkstate to 4
        ;MessageBox "SHOULD BE AT TREE"
    endif

elseif ( walkstate == 4 )

    set timer to timer + GetSecondsPassed

    Fargoth->ClearForceSneak ; terminate sneak mode
    Fargoth->AiWander 0 0 0 0 0 0 0 0

        if ( timer > 3 )
            Fargoth->ForceSneak ; reenter sneak mode
            Fargoth->AiTravel -11410.590,-72057.188,133.644 ;goes to wall
            set walkstate to 5
        endif
endif

```

## Forcing running and jumping: Tribunal NPC Movement Functions



**ForceRun**  
**ClearForceRun**  
**GetForceRun (short)**

**ForceJump**  
**ClearForceJump**  
**GetForceJump (short)**

**ForceMoveJump**  
**ClearForceMoveJump**  
**GetForceMoveJump (short)**

These functions all control the specified NPC's movement. The ForceRun function makes the NPC always run when they move, the ForceJump function makes the NPC constantly jump, and the ForceMoveJump makes the NPC always jump when they are moving. The Get versions of the functions return one if the specified NPC currently is forced into the given action and zero otherwise. The Clear functions are used to turn forced movement modes off. An NPC can only be forced to do one movement at a time. The priority for forced movement is Sneak > Running > Jump > MoveJump.

### Sample Script:

This script lets an object control the movement type of Athlete, an NPC set to Travel endlessly in a four-point square.

```
Begin AthleteControl

short questionAsked
short button
short isrunning
short isjumping

if ( MenuMode )
    return
endif

if ( OnActivate == 1 )
    set isrunning to ( Athlete->GetForceRun )
    set isjumping to ( Athlete->GetForceMoveJump )
    if ( questionAsked == 0 )
        if ( isrunning )
            MessageBox, "Make Athlete stop running? " "Yes" "No"
        else
            MessageBox, "Make Athlete run? " "Yes" "No"
        endif
        set questionAsked to 1
    endif
endif

if ( questionAsked == 1 )
    set button to GetButtonPressed
    if ( button == -1 )
        else
            if ( isrunning == 0 )
                if ( button == 0 )
                    Athlete->ClearForceMoveJump
                    Athlete->ForceRun
                endif
            else
                if ( button == 0 )
                    Athlete->ClearForceRun
                endif
            endif
        endif
        if ( isjumping )
```

```

        MessageBox, "Make Athelete stop jumping? " "Yes" "No"
    else
        MessageBox, "Make Athelete jump? " "Yes" "No"
    endif
    set questionAsked to 2
    set button to -1
endif
endif
endif
if ( questionAsked == 2 )
    set button to GetButtonPressed
    if ( button == -1 )
    else
        if ( isjumping == 0 )
            if ( button == 0 )
                Athlete->ClearForceRun
                Athlete->ForceMoveJump
            endif
        else
            if ( button == 0 )
                Athlete->ClearForceMoveJump
            endif
        endif
    endif
    set questionAsked to 0
    set button to -1
endif
endif
end

```

## Detecting players action: running, jumping, sneaking?



**[no fix] GetPCSneaking (short)**  
**[no fix] GetPCRunning (short)**  
**[no fix] GetPCJumping (short)**

```
if ( GetPCRunning )
```

These functions returns 1 if the PC is performing the appropriate action and 0 if he is not. Since Morrowind doesn't have functions to directly test for keyboard input, these functions provide an alternative to check if the player has a certain button pressed. They have accordingly been used extensively for control purposes, e.g. for movable ships, rideable creatures, or in my climbing mod.

### Sample script:

When this script is placed on an NPC, and the player has an item equipped called “scissors”, MessageBox warnings will be given based on the player’s current actions.

```

Begin momsript
short warn
if ( player->HasItemEquipped "scissors" )
    if ( warn != 1 )
        if ( GetPCRunning )
            MessageBox "Don't run with scissors!"
            set warn to 1
        endif
    endif
    if ( warn != 2 )
        if ( GetPCJumping )
            MessageBox "Don't jump with those scissors! You'll put your eye out!"
            set warn to 2
        endif
    endif
    if ( warn != 3 )
        if ( GetPCSneaking )
            MessageBox "You can't hide those scissors from me!"
            set warn to 3
        endif
    endif
endif

```

```

endif
endif
else
    set warn to 0
endif
end

```

## Detect combat readiness



**GetWeaponDrawn (short)**  
**GetSpellReadied (short)**

```
if ( player->GetWeaponDrawn )
```

These functions can be used to determine whether or not an Actor has their weapon out or whether or not they have a spell readied for casting.

Sample Script: This global script gives notification messages based on the player's weapon and spell states.

```

Begin player_notifications

short weapstate
short spelstate

if ( player->GetWeaponDrawn )
    if ( weapstate != 1 )
        set weapstate to 1
        MessageBox "The player's weapon is drawn."
    Endif
else
    if ( weapstate != 0 )
        set weapstate to 0
        MessageBox "The player's weapon is sheathed."
    Endif
endif

if ( player->GetSpellReadied )
    if ( spelstate != 1 )
        set spelstate to 1
        MessageBox "The player's spell is readied."
    Endif
else
    if ( spelstate != 0 )
        set spelstate to 0
        MessageBox "The player's spell is put away."
    Endif
endif

end

```

**Note:** GetWeaponDrawn will return 0 at some unusual times; when a lock pick or probe is removed through scripting or when it is broken, GetWeaponDrawn will return 0 for a few frames before you switch to Hand-to-Hand. When fighting Hand to Hand, when you equip or remove a piece of armor or clothing, GetWeaponDrawn will return 0 briefly. Thanks to Björn and DinkumThinkum respectively for pointing this out.

## Making someone fall

**Fall**

Actor->Fall

Seems to give an NPC the extra nudge they may need even after you yank the floor out from under them. It also brings down flying creatures. Used by the Icarian Flight guy. When I tried to use this on the player in my climbing mod, it seemed to sometimes "warp" the player directly to the ground below.

## ***Equipment sharing and other companion functions***



**[no fix] companion (is local short)**

```
short companion
Set companion to 1
```

Tribunal introduced the option to "share" equipment with NPC's or creatures. To enable this option, you must define a local short variable named "companion" and set it to 1 or higher. Setting it to 0 or a negative value will disable sharing. This method is used both for mercenaries and for pack animals.

**[no fix] minimumprofit (is local float)**

```
Float minimumprofit
If ( minimumprofit < 0 )
```

Seems to be another variable set by the game, it is probably the difference between current value of all goods and gold minus starting value. If this gets negative, the mercenary can be scripted to quit.

**Sample Script:** here is the relevant part from Calvus' script (the mercenary in Mournhold). This section handles changes in state when Calvus leaves a contract, either because the contract expires, or because the player has taken Calvus' stuff. The sharing is initiated from dialogue (setting companion to 1), not in the script itself

```
if ( GetJournalIndex Merc_Calvus_Quit < 1 ) ;if Calvus has already quit, don't do this
    if ( Contract_Calvus == 1 ) ;if Calvus doesn't have a contract, don't do this
        if ( minimumProfit < 0 ) ;Calvus is quitting because player took his stuff
            AiWander 128 6 0 40 30 20 0 0 0 0 0 0
            Set Companion to 0; stop sharing
            StopScript Contract_Calvus
            Set Contract_Calvus to 0
            ForceGreeting
            return
        else
            if ( Contract_Calvus == 0 ) ;handles Calvus after a contract expires
                AiWander 128 6 0 40 30 20 0 0 0 0 0 0
                Set Companion to 0; stop sharing
                if ( GetJournalIndex Merc_Calvus < 10 )
                    Journal Merc_Calvus 10;first contract expired
                else
                    Journal Merc_Calvus 20; most recent contract expired
                endif
            endif
        endif
    endif
endif
```

**[no fix] StayOutside (is local short)**

```
short StayOutside
set StayOutside to 1
```

A useful variable for companions. When used in a script, it causes whoever it's assigned to to automatically remain (and wait) outside of any interior the player may enter (automatically rejoins upon return). (Forum info / Grumpy)

## Race, Faction and Rank

### *Determining Race*

**GetRace, "RaceID" (returns Boolean/short)**

```
Player->GetRace "Dark Elf"
```

Returns 1 if the object is of the race indicated by RaceID.

**Sample Script:** This is a global script Bethesda uses to set a variable that can be used to determine the PCs race in dialogue:

```
begin RaceCheck

;global script that gets run once to check the PC's race, so it can be used in dialogue

if ( Player->GetRace "Argonian" == 1 )
    set PCRace to 1
elseif ( Player->GetRace "Breton" == 1 )
    set PCRace to 2
elseif ( Player->GetRace "Dark Elf" == 1 )
    set PCRace to 3
elseif ( Player->GetRace "High Elf" == 1 )
    set PCRace to 4
elseif ( Player->GetRace "Imperial" == 1 )
    set PCRace to 5
elseif ( Player->GetRace "Khajiit" == 1 )
    set PCRace to 6
elseif ( Player->GetRace "Nord" == 1 )
    set PCRace to 7
elseif ( Player->GetRace "Orc" == 1 )
    set PCRace to 8
elseif ( Player->GetRace "Redguard" == 1 )
    set PCRace to 9
elseif ( Player->GetRace "Wood Elf" == 1 )
    set PCRace to 10
endif

StopScript RaceCheck

end
```

### *Determining the PC's Faction status:*

**[no fix] GetPCRank, FactionID\_enum (returns short)**

```
if ( GetPCRank "Telvanni" == 9 )
```

Returns PC's rank in faction. This will default to the Actor's faction if FactionID is not defined. Returns 0-9 and -1 if not a member.

**Sample Script:** An Actor/object with the following script is only enabled if the PC is not a member of House Redoran:

```
Begin bandenIndarysScript
if ( CellChanged == 0 )
    Return
endif

if ( GetPCRank "Redoran" == -1 )
    Enable
else
    Disable
endif
End
```

**[no fix?] GetPCFacRep, [FactionID] (returns short?)**

This **doesn't work** in script (causes errors and/or crashes when used).

**SameFaction (returns Boolean/short)**

Returns 1 if PC is in the faction of the calling object (NPC).

**PCExpelled ["factionID"] (returns Boolean/short)**

Returns 1 if PC has been expelled once from calling object's (NPC) Faction, or a faction can be defined to get a specific one. For an example script look below, PCClearExpelled function.

## ***Modifying faction standing and reaction***

**[no fix] PCJoinFaction ["FactionID"]**

Makes the PC a member of the specified faction. FactionID is optional if it is not added it will use the faction of the NPC who called the function

**LowerRank, ["FactionID"]  
RaiseRank, ["FactionID"]**

Raises or lowers the object's rank in its current faction. This function doesn't work in global scripts, but works in targeted and local scripts. It also works in dialogue. You can use the fix to raise a different NPC's rank (NPC\_ID->RaiseRank) in dialogue results **only** - in scripts this function will only work on the actor the script is attached to.

The faction ID argument is totally ignored by the game engine, so is not required, and even if you have the faction ID as something totally irrelevant (for example "eel pie"), the script still compiles and runs fine.

You cannot use RaiseRank to have an NPC join a faction (unlike the PC version) - if the NPC doesn't already belong to a faction RaiseRank has no effect.

**[no fix] PCLowerRank, ["FactionID"]  
[no fix] PCRaiseRank, ["FactionID"]**

Raises or lowers the PC 1 rank in the NPCs faction. If PC is not part of the faction, it will set the rank to 1.

### **Example Script:**

```
Begin treboniusScript
;if you're both in the Arena
;and if the guildmaster quest is active...
;sets a journal and raises the player's rank
;when trebonius dies.
short doOnce
short nalore

if ( doOnce == 1 )
    Return
endif
if ( GetJournalIndex MG_Guildmaster < 50 )
    Return
endif
if ( GetPCCell "Vivec, Arena" == 0 )
```

```

        Return
    endif
    if ( duelActive == 0 )
        Return
    endif
    if ( OnDeath == 1 )
        Set DuelActive to 0
        Set doOnce to 1
        PCRaiseRank "Mages Guild"
        PCRaiseRank "Mages Guild"
        Journal MG_Guildmaster 100
    endif
End

```

### [no fix] PCExpell ["FactionID"]

Expels PC from NPCs faction.

### [no fix] PCClearExpelled ["FactionID"]

Clears "currently expelled" flag on the player.

### Example script:

A script by Bethesda, which clears the Players expelled status after some time:

```

Begin expelledMG

;this is just a model
;it is supposed to be on an item in each of the Mages Guilds.

short myDay
short temp

if ( PCExpelled "Mages Guild" == 0 )
    return
endif

if ( ExpMagesGuild == 0 )
    Set ExpMagesGuild to 1
endif

if ( myDay == 0 )
    Set myDay to Day
endif

if ( myDay == Day )
    return
endif

if ( Day > myDay )
    Set temp to ( Day - myDay )
else
    Set temp to ( myDay - Day )
endif

Set myDay to Day

Set temp to ( temp + 2 )

Set ExpMagesGuild to ( ExpMagesGuild + temp )

if ( ExpMagesGuild > 30 )
    Set ExpMagesGuild to 0
    PCClearExpelled "Mages Guild"
    return
endif

End

```

```
[no fix] ModPCFacRep, var_enum, ["FactionID"]
[no fix] SetPCFacRep, var_enum, ["FactionID"]

ModPCFacRep, 5, "Imperial Legion"
ModPCFacRep, 5, "Temple"
```

Modifies or defines the reaction modifier for members of the specified faction (towards the PC).

```
ModFactionReaction, "factionID1", "factionID2", var_enum
SetFactionReaction, "factionID1", "factionID2", var_enum
```

Modifies or defines the reaction of one faction towards members of another faction.

**Example:** This is part of the MoonAndStar script. This part first makes the PC part of the faction "Nerevarine" and then sets two factions to react particularly to this change:

```
;faction reaction and journal stuff
Journal "A2_6_Incarnate" 50
player->modReputation 5
PCJoinFaction, Nerevarine

if ( GetPCRank, Redoran >= 0 )
    modFactionReaction, "Redoran", "Nerevarine", 4
endif

if ( GetPCRank, Temple >= 0 )
    modFactionReaction, "Temple", "Nerevarine", 4
endif
```

## Determining and changing reputation and disposition

### Get/Mod/SetReputation

### Get/Mod/SetDisposition

SetDisposition refers to base disposition (as set in the TES CS, unaltered by any modifiers). To set the disposition value used in game, use a combination of GetDisposition (returns current value) and ModDisposition; for example, to completely control an NPC's disposition from a global "myRealDisp" (can be a short, it doesn't need to be a float unless you want to change disposition by float values), the NPC's local script could include something like this:

```
if ( myRealDisp > 100 )
    set myRealDisp to 100
elseif ( myRealDisp < 0 )
    set myRealDisp to 0
endif
set localFloatVar to ( GetDisposition )
set localFloatVar to ( myRealDisp - localFloatVar )
ModDisposition localFloatVar
```

## Werewolf-specific functions

### Set the werewolf attributes



#### SetWerewolfAcrobatics

Actor->SetWerewolfAcrobatics

This function set the attributes of the object to those of a werewolf. This sets the targets skills and attributes to match the fWerewolfxxxx gameplay settings. In most cases, this means a high strength, agility, acrobatics etc, and 0 in most other things.

```
Player->AddSpell "werewolf vision"  
Player->AddSpell "werewolf regeneration"  
Player->SetWerewolfAcrobatics
```

### Change the color of Secunda



[no fix] TurnMoonWhite

[no fix] TurnMoonRed

These two functions are very simple – they change the color of Secunda (The small, white moon) from white to red and back again. This doesn't have any real effect to gameplay, but it does make the sky look different. It is used during the Bloodmoon main quest – hence the expansion title.

```
if ( doOnce == 0 )  
    TurnMoonRed  
    set doOnce to 1  
endif
```

### Determine how many kills a werewolf has



[no fix] GetWerewolfKills (returns short ?)

This keeps count of how many NPC's killed by the player when in werewolf form. Each time an NPC is killed while the PC is a werewolf, one is added to this count. It is reset automatically when the PC changes back into human form.

```
if ( GetWerewolfKills > 0 )  
    ; Do code to stop the PC from being affected by the hunger.  
endif
```

## Check to see if the creature is in werewolf form



### IsWerewolf

```
If ( Actor->IsWerewolf )
```

This function allows to determines if the target is a werewolf or not. It can be used on the PC or other creatures.

```
if ( Player->IsWerewolf != 1 ) ;DON'T RUN IF PLAYER ISN'T WEREWOLF
    return
endif
```

## Change to a werewolf



### BecomeWerewolf

### UndoWerewolf

```
Actor->BecomeWerewolf
Actor->UndoWerewolf
```

These functions change the target object to a Werewolf or change them back to their original form. IMPORTANT: Using Becomewerewolf and Undowerewolf CAN break your game. Some quests and variables depend solely on use of these, so if you use one to toy around.... you may be asking for it. (This message brought to you by your friendly dev WormGod).

Note:

When the player is a werewolf, the only thing you can detect them activating (Using OnActivate) is doors.

If you want to detect the player activating an object, or even another werewolf, one method is to use an invisible door and place it over the object you want to detect the activation of.

```
if ( OnPCEquip == 1 )
    Player->BecomeWerewolf
    Set OnPCEquip to 0
Endif

Set timer to ( timer + GetSecondsPassed )

If ( timer > 10 )
    Player->UndoWerewolf
Endif
```

## Special werewolf global variables



**[no fix] PCknownWerewolf (is short global)**

A global variable that indicates whether the PC is a known werewolf



**[no fix] PCWerewolf (is short global)**

Set to 1 when player is werewolf. Used in controlling numerous werewolf scripts.

It works in the same way as PCVampire:

-1 = Player has been cured of lycanthropy and cannot become a werewolf again

0 = Player is not a werewolf

1 = Player is a werewolf



**[no fix] WerewolfClawMult (is float global)**

A multiplier for claw damage. Exact formula unclear, see werewolf scripts for reference.

# Text and Dialogue

## ***Brief dialogue how-to***

Dialogue is an art in itself that can not be fully covered here. However, when creating quests you will often mesh dialogue and scripting to control your quest and achieve certain effects. Therefore, I will give a brief introduction.

## **The concept of MW dialogue**

In Morrowind, dialogue is organized in a database. This database is structured in the following way:

### **Top level:**

#### **The different "divisions" of dialogue are:**

*Topic:* The actual topic words and responses for the dialogue window in the game.

*Greetings:* The text strings you are greeted with when you initiate dialogue with an Actor.

*Persuasion:* The answers you get for successful or failed persuasion attempts.

*Journal:* The entries for your journal.

*Voice:* The .mp3 sound files (and subtitles) that players "say" when you come near, are hit, are fleeing, etc.

Find these divisions on the tabs on the top left of the dialogue window (use the arrows to get to the hidden tabs).

### **Sub-level 1:**

Each of these top level divisions has sublevels I call topics – for the *Topics* division, these are the actual "keywords" (topics) that Actors will have something to say about. These are the words that will be highlighted ("hyperlinked") in in-game text and listed in the right hand panel of the dialogue window. For *Journal* these are the different journals (usually one per quest). For *Voices* the different categories of sound responses that are "triggered" by the appropriate in-game conditions etc. For *Greetings* these are just general categories of greetings (diseased, quest offers, standard and so on). This is the way Bethesda has used them (forum info / Emma):

- Greetings 0: Npc is alarmed
- Greetings 1: Quests (actually quests where it doesn't matter if the Player is a vampire, is nude, is a criminal, is diseased).
- Greetings 2: Player is a vampire/player is nude
- Greetings 3: Traitors to Morag Tong
- Greetings 4: Crime and disease
- Greetings 5: Quests
- Greetings 6: Factions
- Greetings 7: Classes, Endgame, Slaves
- Greetings 8: Clothes (general greetings concerning how player is dressed)
- Greetings 9: Locations.

For *Persuasion* these are things like service refusals or bribe success/fail messages.

### **Sub-level 2:**

For each topic there can be one or several sub-entries ("Info/Response") – these are the actual responses. For *Topic* and *Greetings* these are the actual answers of Actors to the topic phrase. For *Journal* these are the different entries describing the progress of a quest.

These entries represent a linked list, meaning that each entry contains (invisible) info on which is the next response in line and which is the previous. (This leads to a common error message when dialogue gets deleted or scrambled, e.g. by cleaning a mod with TESAME, or by loading several mods that change the same topic. The game doesn't know for sure any more which entry comes after which. This sometimes doesn't matter, but it can also mean that certain responses are cut off, because the order isn't right any more – this depends on the conditions). To check and/or change the next/previous ID strings, you can export the dialogue and re-import it if necessary (solution suggested by Kateri).

Dialogue entries come with conditions that you can set in the dialogue editor window. In the topic window you will find a list of general conditions on the left – here you can define which Actor (Actor ID) or group of Actors (Race, Class, Disposition, etc.) potentially know this response. There are also two conditions for the PC (PC faction and rank). To the right you can set a maximum of 6 "free" conditions that can refer to the Actor, the PC or other things like the state of global variables – lots of options here, you will have to see for yourself. Check for journal entries, player stats, local or global variables, items in inventory, and many other functions, some equivalent to script functions, some unique to dialogue (see below).

An important and initially confusing feature of the dialogue editor is the filter option (lower left-hand corner). When you select an Actor ID here, you only see the topics that this Actor can possibly know (as described above). Remember that when you create a **new** topic (maybe specifically for that NPC) it contains no responses. Thus the Actor cannot "know" it and thus the freshly created topic does not appear! Select the empty slot on the very top to see all topics again, make a response for the new topic that your Actor can "know" – then you can turn the filter on again, if you wish.

## How dialogue works

To decide whether a dialogue **topic** is available during dialogue with an NPC, the game checks:

1. Whether the NPC "knows" the topic – this is determined by the conditions set in the "speaker conditions" field – if the NPC can fulfill the conditions for just one of the responses for that topic, he "knows" it.
2. Whether the Player knows it. The PC knows a topic if it (the topic word) either appeared previously in a response from an NPC or if it was specifically added with the AddTopic function. Note that the addition of a topic to the list of topics the PC knows only happens if the keyword becomes available in the same NPC's dialog that says the topic keyword. An example: let's say you try to trade with a weapon merchant and he denies you service because you have skooma in inventory. The word "skooma" is used in the dialogue, and there is a topic "skooma", but it is not familiar to weapon merchants. After that, you talk to a savant who *is* familiar with the topic... but you are still not. The topic will appear in your list only when he (or someone else familiar with it) will mention skooma in one of responses (Forum info / Kir).
3. If both the PC and the Actor "know" the topic it shows up in the topic list and gets highlighted in the text – it can now be selected by the player and an appropriate response will be given.

When the game has to select the correct response from the list of responses for that topic, it does the following:

- It starts checking **from the top** of the list, whether the conditions for that response are met, which means **all** conditions that have been defined for that response return "true".
- If not -> Game moves on to the next item in the list and checks again.

- If yes -> this entry is used and printed to the dialogue window.

Special rule for greetings: if none of the items in the list are met, move on to the next level 1 item (e.g. if no greeting in "Greeting 0" returns true, start checking "Greeting 1").

Special rule for Journal: there is only one condition here, the index (called by `Journal` function). It must be exactly met.

Once a response has been selected, the game will

- Output the text string to the dialogue window (or play the mp3 for *Voice* responses)
- Execute any Script commands in the Results field (at the very bottom). You can use all scripting functions here. Conditions (*if* command) can also be used (Thanks to Kir and Manauser for this info). Nested ifs are possible with Bloodmoon, not sure about earlier versions.

Be aware that the result field allows you to exchange information with scripts (e.g. by setting variables or adding journal entries) and that scripts can vice versa influence dialogue as well, by setting conditions that can be tested (e.g. you can check for local or global variables in the speaker conditions of dialogue). The simplest script that influences dialogue is the *nolore* script, which is just used as a flag to keep Actors from using standard dialogue.

**Note:** Result field scripts are not compiled by CS. You may write any rubbish there and MW will not complain until the line is told by an NPC and its script is compiled on-the-fly. If the script is complex enough to worry about possible syntax errors, it is recommended to copy/paste its text into a regular dummy script and try to save it. That is more reliable than using the "Error Test Results" button, as it can change preset values of global variables. On the other hand, the fact that resultbox scripts are compiled at runtime may allow some interesting effects, like addressing contents of another mod without duplicating it in the current mod, which can't be done with conventional scripts (Forum info / Kir).

## Creating clean dialogue

*(Thanks to JOG and Cyran0 for this information)*

*When you are adding dialogue to an existing topic (including Greetings, Voices, etc):*

- Be very careful in using a copy/edit based method. **Never** change an original line: only edit the copy (top line of the two). Editing the copy is safe, but the two may be easily confused.
- **It is preferable to "clean" the original dialogue lines (infos) on either side of the inserted dialogue.** This can be done in "Details" view by toggling the "ignore" flag for the changes, or by using a third-party tool, as outlined in the 'Cleaning up your mod' section.
- It is strongly recommended that you use only one insertion point wherever possible (you can usually achieve this by duplicating the text of original lines in new locations if need be).
- If you are removing **all** entries you have added to a topic, or cleaning original dialogue of changes: use the details view, TESAME or a similar utility that will *remove your changes*.
- If you are removing one or more lines you have added (but not all lines in a block), do not simply clean the changes. In order to ensure the info-IDs are updated correctly, you should *delete the line(s)* in the dialogue editor itself (right-click and delete, or 'delete' key when line is selected). The editor will update the IDs automatically. (If for some reason you cannot do this, you can move the dialogue line to be deleted as far

away from all other new dialogue as is possible, then use ignore: moving the info will also update IDs automatically, and the original dialogue modified by the move may be cleaned as above. Alternatively, you can export the dialogue, edit as necessary and re-import it.)

### **Some explanation:**

Dialogue is a linked list: each info contains its own ID-string, and the IDs of the previous and next lines. You can see this more easily by exporting the dialogue and viewing it in a text editor: the long numbers are the IDs, and the pattern should be obvious.

When you change an original line, the ID-string stays the same: this has a similar effect to the one modders are familiar with in different mods changing the same thing, or adding objects with identical IDs... *The last mod to load wins*. The text and conditions of the last mod to load will overwrite the text and conditions in other mods, dialogue from mods that load earlier and contain that line may lose its place, and some entries may be shifted to new (and usually highly problematic) positions in the list. (New entries themselves are highly unlikely to be subject to the same problem, as 'copy' or 'new' creates a new, randomly-generated ID for the new dialogue line: note that 'copy' does not copy the ID, it generates a new one. However, if you were to copy an .esp with dialogue, e.g. in order to reuse the dialogue for a different NPC, the dialogue would still have the same info-IDs as in the original mod and the two mods would be incompatible. This still applies if you change the topic IDs: you need to change the info-IDs as well!)

When you add dialogue to an existing topic, this in itself modifies the original dialogue around it as the CS updates the next/previous info-IDs to point to the new line. That changed original line can overwrite (or be overwritten by) the same line - *including its links to the next/previous lines* - in other mods when the game loads, causing dialogue lines to lose their place and be moved.

The way to avoid this (as much as is possible) is to clean the changes from the original dialogue. This will give errors when the mod is loaded in the CS ("previous/next string is different than expected for ID..." kind of errors)... but since the new dialogue lines still have the correct previous/next IDs the dialogue will (if all mods are clean) be correctly placed when loaded in either the CS or the game, and since the original dialogue is not changed it will not overwrite original lines for other mods. Problems can still arise in some circumstances, most particularly if an "unclean" mod loads later, but most modders will already be familiar with this problem in other areas (object property editing, for example, can be similar).

One problem which can arise with this method is that if your dialogue uses more than one insertion point, so that an original line has new dialogue both directly above and directly below, cleaning it will cause problems. The safer method is to simply copy the original line and use the copy, leaving the original below all your dialogue (and therefore unused by the game) - thus keeping a single insertion point.

### *Further notes:*

Another point which I think fits the spirit if not quite the letter of 'creating clean dialogue' is this: It is not advisable (or, perhaps, not *polite*) to put your more generic dialogue higher in the chain than it needs to be, nor to filter any of your dialogue more broadly than necessary. This just leads to a game of one-upmanship as modders try to get their dialogue higher to prevent other mods breaking it.

If possible, avoid adding dialogue that will be used by existing NPCs (those not added by your mod) - and be especially careful of this in Greetings. It is often suggested that you use filters unique to your mod when creating new dialogue that is not filtered for actor ID, e.g. new factions, classes, or cells. While this is good advice in general, be aware that unless you prevent it, many players are likely to bring companions into your new cells - and those companions will have any dialogue that is filtered solely for that cell (usually not a desirable outcome). If possible, you may want add a nolore filter to prevent this (preferable to using the companion variable as some still use Morrowind-only companions, and almost all companions and other followers are nolore'd - but "not local companion >= 0" will still rule out the vast majority of companions). You can also filter dialogue for an otherwise-unrelated selection of NPCs from your own mod by using a uniquely-named local variable as a filter (but all those NPCs will need to have local scripts for this to work).

## A few golden rules

- **The most specific responses should be on top of the list, the "catch all" answers should be lowest!** Remember the first one that returns true is the one that gets picked. So you can't have a response for everyone in Vivec above one for a specific NPC in Vivec.
- **If you want an NPC to be able to talk with the PC about something special, you must introduce the topic word, e.g. in a greeting or in a "latest rumors" response.** Alternatively, you can use a script with the AddTopic function.
- **Don't use normal words as journal topics.** Topic, Greeting, Journal are actually all in the same database - that's why journal topics use a format like A1\_dreams. If it were just "dreams" than the journal entry to "dreams" could come up as a dialogue response to the word "dreams".
- **Never delete a topic that belongs to the original Bethesda master files.** This is very hard to repair and will cause severe errors in peoples save games. (Emma). You should also avoid deleting responses from the original game: the best way to remove a response is simply to add your custom dialogue above the original response, so it still exists unchanged but is never "said"; if this is not possible, you can change the filtering for the response (e.g. filter for ID "dialog placeholder" will remove the response from all NPCs in the game world) - but changes to an original response will be overwritten by any later-loading mod that also changes that response.
- **If you are using the greetings section 1, don't put your greetings at the very top of it.** The top greeting belongs to a certain quest, and must be left at the very top in order to always show up. You can put your greetings below these instead. (Emma)
- If at all possible, avoid putting your new responses as the very first or very last entry in an existing topic (this includes greetings). For example, if you want your greeting to be above the first greeting in Greeting 2, put it in Greeting 1. If you want it below the last greeting in Greeting 1 as well, then copy that greeting and put the copy above your new greeting: this will achieve the same effect, as the original line will never be "said".

## Dialogue 101

The following summarizes some of the most frequent problems with dialogue. This list was assembled from a forum discussion with contributions by Klinn, Emma and GarryB.

Tip 1) **My new topics disappear!** Go to the Filter box at the bottom of the list of topics. Clear the filter by choosing the top empty line in the drop-down list. Recommend using the button

on main toolbar to bring up the dialogue editor rather than from the NPC's properties (if you open the dialogue window from an NPC's property box, it will be filtered for that NPC by default; but if you open the window from the main toolbar or menu, it will be unfiltered by default).

Tip 2) **My NPC keeps asking me a question over and over!** Be sure to put the replies *above* the original question. Sounds backwards, but it works.

Tip 3) **My NPC talks about everything!** To keep an NPC from having the standard topics about Morrowind lore, attach the script "NoLore" to him or her. If you already have a script on the NPC, add the declaration **Short NoLore** near the top.

Tip 4) **My NPC still has extra topics!** Some other general topics may appear depending on an NPC's faction or class. For example, members of the Imperial Legion will always automatically have topics about that faction, the Empire, and more. There are some lore topics that are not well-filtered and will appear in certain circumstances regardless (especially with Bloodmoon: many BM NPCs are NoLored to avoid *Morrowind* lore topics, and most Bloodmoon lore topics are filtered only for cell). If this is a real problem, and your mod design allows for it, you can use a creature instead of an NPC: creatures can only have dialogue filtered for specific ID.

Tip 5) **How do I add topics for just my NPC?** After creating the topic and it's Info/Responses, in the Speaker Conditions area, set the ID to your NPC.

Tip 6) **I added topics but my NPC doesn't have them!** Two possibilities: the PC must have already heard (read) that topic word or phrase before he can ask about it. Usually this is done by having the an NPC's greeting include the topic. Second possibility: there may be Speaker Conditions that prevent the topic from appearing. Even if it appears when you filter the dialogue for an NPC, some topics depend on the player having reached a certain point in the game, having a specific journal entry, and so on. Note that since the resultbox scripts are executed *after* the display has updated, a topic made available by a resultbox script may not appear immediately. Another little trap is that topics won't be added and hyperlinked in dialogue if they don't begin with a letter, so topics like "-follow" won't be hyperlinked in a greeting even if the conditions are met: you will need to use AddTopic.

Tip 7) **How do I change the order of my Responses?** Use the left-arrow and right-arrow keys to move an Info/Response up or down in the list. Note that every original entry that is "touched" by your moved dialogue will be marked as changed: you can remove your changes using the details view.

Tip 8) **How do I create dialogue for creatures?** Any creature can have its specific dialog. You do this exactly as you create the dialog for an npc, with one difference. You have to have the dialog UNFILTERED when making the dialog (i.e. the slot below the topics must be empty). Once you have created the dialog lines, you can filter them for your creature. *Note: This is because creatures can only have dialogue that is filtered for specific ID, and when you create a new response it isn't filtered at all - so the creature can't "know" it.*

Tip 9) **What are typical uses for the dialogue result box?** Emma lists these useful and frequently used commands for the result box:

- Player->AddItem "my item" 1 (a specific item is added to players inventory)
- Player->RemoveItem "my item" 1 (a specific item is removed from players inventory)

- `ModDisposition 5` (npc will like player 5 points better)
- `cast "my_new_spell" player` (the npc will cast a certain spell)
- `AiFollow Player 0 0 0 0` (npc will start follow player)
- `AiWander 0 0 0 0 0 0 0 0 0 0 0` (npc will quit following the player)
- `SetFight 100` (npc will start attacking the player)
- `StartCombat player` (npc will start attacking the player)
- `StopCombat` (yep, you've guessed it. Stop combat)
- `StartScript "my_global_script"` (start a certain script)
- `Set companion to 1` (if you have added a "short companion" command to your script, this will make the npc share with you; requires Tribunal or Bloodmoon)
- `SetHealth 100` (will set the npc's health to 100 - same command can be used for setting other skills and attributes as well, i.e. `SetMagicka`, `SetLongBlade` etc.)
- `disable` (will make the npc instantly disappear)
- `goodbye` (will force the player to end the conversation. Can be useful for instance in order to avoid further small talk with a npc that has already been disabled )

Tip 10) **I have created so much dialogue, how can I possibly spell-check it?** Checking spelling and grammar can be streamlined by using the export and import functions in the Construction Set. Export "new" dialogue to a file, use your favorite editor for automatic spellchecking and corrections and import the corrected dialogue . Much easier on the brain than jumping around in a myriad of topics, greetings and journal entries. (Forum info / GarryB)

## Dialogue-related functions

Many of the following functions are not only used in scripts, but also in the result field of the dialogue editor window.

### Displaying messages

**[no fix] `MessageBox, "Message", [var1], [var2], ["button1"], ["button2"]`**

The *MessageBox* command lets you give out information to the player. Normally these appear as a small box with the text on the bottom of the screen that stays there for a few seconds or until the player has clicked a button if the message box has buttons. There is a limit of 9 buttons per message box. If a dialogue window is open, *MessageBox* will output to the dialogue window! This will be in a different color so it's a good way to show that text isn't part of dialog. For example, "Okay I'll take the curse off. *He takes the curse off.*" *MessageBox* has several different modes of operation. The simplest one is just giving out an onscreen message that appears on the bottom of the screen for a few seconds, as in the following script that gives out a message when the item it's attached to is equipped:

```
Begin informplayer
Short OnPCEquip

if ( MenuMode == 1 )
    return
endif

if (OnPCEquip ==1 )
    MessageBox, "The sword vibrates in your hand"
    Set OnPCEquip to 0
Endif

End informplayer
```

The second mode of operation makes the message stay on the screen until the player presses a button:

```
MessageBox, "Ulyah lifts her hands and speaks the formula. You will now be transported to  
Sheogorad", "ok"
```

In the third mode of operation you can use the messagebox to demand a decision from the player via a message box with buttons and the *GetButtonPressed* function.

**Warning:** The use of more than one messagebox in a frame can cause a CTD. This also includes the “Your journal has been updated” messagebox.

Multiple MessageBox commands in the same frame can cause a crash. The problem doesn't seem to appear if you do 2 MessageBox commands in a row, but stick any other commands between them and you'll have problems. Using 3 simple messages in a row can work, but usually crashes if there are variable substitutions involved.

I'm guessing it's partially an interaction with the voice subtitle boxes. There may be some overall limit to the number of messages per frame. -(CDCoolley)

Notes by DinkumThinkum:

1. My impression is that any type of text messages have the potential for triggering a crash if they're generated in the same frame.
2. The bug doesn't trigger a CTD every single time you generate two text messages in the same frame: it appears to be a random chance of it happening. But, from what I saw when I tested this: if you keep generating two or more text messages in the same frame (with some other code in between them), sooner or later you *will* crash. Sometimes my test script would cause a CTD within a few seconds, sometimes it would take five, ten, or more minutes, but it would invariably crash if I waited long enough.
3. For a start-up script I was working on when I ran into this, adding a one frame delay between popping up a Message Box message and updating the player's Journal eliminated intermittent start-up CTDs I had been getting.

There also appears to be a problem if you call too many message boxes from the same script, even if you set up a frame counter. I'm not sure the limit, but I discovered it when testing NecroRise a while ago... Apparently, 50 is too much. (Forum info / Cid88)

**Note:** If you use a Tribunal start script to give out a message box with a button as soon as the game is loaded, you should delay the MessageBox, otherwise the mouse pointer will not be displayed.

A one frame delay isn't enough if you have a massive start-up script that runs as soon as the game is loaded. So you might want to delay Message Boxes (that have buttons) for a second or so after the game is loaded, just to be on the safe side. (Forum info / DinkumThinkum).

If there is already a messagebox with choices on the screen, opening another choice messagebox will cause the first choice messagebox to vanish. However, when the player picks on of the choices, the messagebox vanishes and the crosshair appears but the player can't move.

Using menutest, 0 will allow the player to move again.

You can enter carriage returns to message boxes, but it requires hex-editing the .esp. Put some unusual characters like ||, then save the esp. Then hex edit the || characters and make them 0D0A (hex for carriage return). (Forum info / qarl)

### [no fix] GetButtonPressed (returns short)

Pressed button if a message box with buttons is used, starting at 0. Will return -1 until button is pressed.

#### Sample Script:

```
Begin choices

Short button
Short status
Short OnPCEquip ;declare as variable - otherwise there will be errors

if ( OnPCEquip ==1 )
    MessageBox, "The sword vibrates in your hand. Do you want to equip it?", "yes", "no"
    Set OnPCEquip to 0 ;display the Message Box only once
    Set status to 1
Endif
If ( status == 1); wait for player decision
    Set button to GetButtonPressed
    If ( button == -1 ); no button selected yet: do nothing
        return
    ElseIf ( button == 0 ); continue normally
        Set status to 0 ; reset for next time
    ElseIf ( button == 1 )
        Player->drop, "power_sword" ; makes the player drop the item
        Set status to 0
    Endif
Endif
```

End

## Displaying variables and text defines in a message box

To display variables in a message box you need to use a syntax describing the format of the number to be displayed. ATTENTION – there is a lot of wrong info in the original helpfile on this!

```
MessageBox "You have %.0f days left", days_left
```

The % symbol indicates the variable. The number after the dot determines the number of digits displayed. "f" signifies a float variable. The helpfile lists several types (f for float, D for short or long and S for string variables), of these I could only get f to work. However %g and %G work fine for short and long variables (thanks Niyt Owl). You can use things like %.3g, but the digit designation will simply be ignored. The designators are not really specific to the variable type, %.3f will also display a short or long variable. There is a limit of 9 variables that can be displayed in the same messagebox (though for some reason the error message for exceeding this limit is "Max variables of 10 exceeded on line XXX").

String variables are mentioned in the helpfile but are to my knowledge not implemented, you can however use dialogue text defines in message boxes but do NOT use %: for text defines – In scripts it's ^instead (thanks Ragnar\_GD):

#### Text defines:

**^PCName**      The player's name.  
**^PCClass**     The player's class.  
**^PCRace**      The player's race.  
**^PCRank**      The player's rank in the speaker's faction.  
**^NextPCRank** The player's next rank in the speaker's faction.

**^Cell**        The cell the player is currently in.

**^Global**      Any global variable value. Floats display as 1.1, such as  
**^GameHour**

**Note:** you can also display a Global variable normally, using the above syntax such as %.1f, which would yield the same result. If you use the ^Global text define in a book, Morrowind will usually crash if you access or change the global variable while the book is open. This should be avoided at all costs! (Forum Info/Chris\_K)

**^Name**          The speaker's name.  
**^Race**          The speaker's race.  
**^Class**        The speaker's class.  
**^Faction**      The speaker's faction. If they have no faction, it will be blank.  
**^Rank**          The speaker's rank.

**Note:** These last listed ones will not work quite as they do in dialogue, as the defines default to the PC's values by default, not to the calling Actor. So ^Name and ^PCName will both display the PC's name, even if the MessageBox is called from dialogue results.

**Example Script:** Stupid sample script demonstrating all possible syntax:

```
Begin test1
short var_1
long var_2
float var_3
; GameHour is a global float variable

set var_1 to 1
set var_2 to 2
set var_3 to 3

MessageBox "PCName, you have %g head, %G hands, and %.5f gold. One could say the hour is
getting late in ^cell. It's the ^GameHour hour or more exactly the %.2f hour!", var_1, var_2,
var_3, GameHour

End
```

## Adding a dialogue topic

[no fix] **AddTopic, "Topic"**

AddTopic, "Topic"

Once you have set up a dialogue topic in the TESCS, you may find that you still can't talk about it with the NPC you have given the dialogue to, because for the game you don't know that particular topic yet. There are two ways to change that condition: either you introduce the topic in another conversation topic (e.g. a custom greeting) or you give it to the player via script, which makes sense when it's an obvious topic the player would ask about without being brought to it by conversation (e.g. if you see an NPC standing under a waterfall, you might want to ask him about "aren't you getting wet?" even if the NPC doesn't bring up the topic.

To do that, just attach a small script to the NPC:

```
Begin AddSpecialDialogue

;add possibility to ask about travel
AddTopic, "aren't you getting wet?"

End AddSpecialDialogue
```

**Note:** You must already have the topic with this topic ID set up before you make this script, otherwise the script compiler will complain.

Even if a topic is introduced through dialogue, it may also be useful to use AddTopic in the resultbox of the line that introduces the topic. If another mod adds a topic with the same ID or an ID of which your topic's ID is a subset, your topic may not be hyperlinked, but AddTopic will ensure that it appears in the speaker's topics list anyway.

AddTopic adds the topic to the **player's** "known topics" list. So using "Actor\_ID"->AddTopic "blabla" is wrong: an NPC's known topics are entirely predefined by speaker conditions.

You can not remove a topic via script; you can however set a speaker condition in the Dialogue editor which can be set from script (e.g. a variable or journal entry), which can be used to achieve the same effect.

## Initiating and ending dialogue

### ForceGreeting

ForceGreeting can be used to make Actors initiate dialogue. When *ForceGreeting* is called the dialogue window will open, and the Actor will use a greeting according to his dialogue settings. Therefore, if you want a special greeting by the Actor, you have to provide it via the dialogue window in the TES CS. It does not matter where the NPC is, this function will always work, so its usually best used in connection with a *GetDistance* or *GetPCCell* condition.

Note that ForceGreeting will **not** work remotely if the player has not encountered the NPC in the last 72 game hours, **unless** the NPC has "corpses persist" checked (in which case it will work as long as the player has encountered the NPC at some point during the game). This also applies to the "Talked to PC" flag. -(Forum info/Neko)

An alternative workaround: If you use PositionCell on the NPC once per day (even without changing their location), the 72 hour time limit no longer applies (Forum info /Time limit info from Cortex, thanks to Srikandi for bringing it to my attention). This trick to get around Actors breaking their connection to you after 72 hours seems to require the cell you send them to to not be the cell where you initially met them (Forum info / Cortex). This either implies it must not be their editor starting cell or that it must be a cell that you have not visited. In my fix I have an interior I send them to for this purpose so either explanation could be why it works. So basically after you have met them they get sent there each day even though they are already there after the first sending.

See the "72-hours bug" section under "Tips and Tricks" for more information.

Using ForceGreeting *in dialogue results for a greeting* can be a useful trick in some circumstances, usually to have a resultbox script executed without providing extra, unique text for it (you can just use a dot "." as the text). The NPC will then give whatever greeting would normally be given, with just an easily-ignored dot above as evidence that your resultbox script was executed. *Note that you **must** change the tested conditions in the resultbox script - otherwise it will loop continuously and crash the game!*

Note also that if an NPC's "Talked to PC" flag is set by your "fake" greeting, any normal greetings that rely on it won't be given. If this is an issue, you could filter your fake greeting for "Talked to PC != 0" to avoid this (the player will then have to talk to the NPC a second time before your script runs).

**Example Script:** this script shows a nice set of condition being checked before initiating the *ForceGreeting* command

```
Begin balynScript

float timer
short doOnce

if ( GetJournalIndex "DA_Mephala" < 40 )
    Return
endif
if ( GetJournalIndex DA_Mephala >= 60 )
    Return
endif
Set timer to ( timer + GetSecondsPassed )
if ( timer < 5 )
    Return
endif

Set timer to 0

if ( doOnce == 0 )
    if ( GetDistance Player <= 1024 )
        if ( player->GetDistance "hlaalu_loaddoor_ 02_balyn" <= 256 )
            if ( GetLOS Player == 1 )
                ForceGreeting
                Journal DA_Mephala 55
                set doOnce to -1
            endif
        endif
    endif
endif
endif

End
```

**Note:** If you use ForceGreeting from within an if block, the script will continue to execute the remaining elseif/else tests instead of skipping over them (as it should). To avoid this, add a *return* after the ForceGreeting.

This script will print the messagebox:

```
if ( ( Actor_ID->GetDisposition ) >= 80 )
    Actor_ID->ForceGreeting
else
    MessageBox "I'm not talking to you!"
endif
```

Here is the corrected version:

```
if ( ( Actor_ID->GetDisposition ) >= 80 )
    Actor_ID->ForceGreeting
    return
else
    MessageBox "I'm not talking to you!"
endif
```

## Goodbye

*Goodbye* forces the end of dialogue: after calling this function the PC can only choose the goodbye option and thus close the dialogue window. Usually this function is used in the result section of a dialogue topic, not in scripts; however, it can be used in scripts if the dialogue window is open.

## Allowing forced Dialogue with Werewolf player (Bloodmoon)



## [no fix] AllowWereWolfForceGreeting (is short variable)

Short AllowWereWolfForceGreeting

This variable allows to use ForceGreeting on a werewolf character. Used on cariushuntscrip, dulkscript, heartfanghuntscrip. The variable simply has to be declared, not set to a specific value.

### Example:

```
Begin dulkScript

short doOnce
short playerwolf
short AllowWereWolfForceGreeting

if ( GetJournalIndex BM_FrostGiant2 == 10 )
    if ( doOnce == 0 )
        if ( GetDistance Player <= 512 )
            if ( Player->IsWerewolf == 1 )
                ForceGreeting
                set doOnce to 1
            endif
        endif
    endif
elseif ( GetJournalIndex BM_FrostGiant2 == 70 )
    if ( doOnce == 1 )
        if ( GetDistance Player <= 512 )
            ;if ( Player->IsWerewolf == 1 )
                ForceGreeting
                set doOnce to 2
            ;endif
        endif
    endif
endif

End dulkScript
```

## Multiple choice – asking questions

[no fix] Choice, "choice 1", choice1\_enum ["choice 2", choice2\_enum, ...]

Choice "yes", 1, "No, certainly not!", 2

This is used in dialogue result fields to ask a decision of the player or can be called just to "continue" a longer speech. After the PC makes his choice, the same topic will be checked again, and you can provide the correct response by using function / choice / = / choice\_enum in the speaker conditions of the dialogue window. Choice can be used in script, if the dialogue window is open. If the same resultbox/script contains more than one choice call, the choices are presented to the player as a single list (unlike MessageBoxes).

The limit of choices per result or per call may well be version dependent. It has been reported that there is a limit of 5 choices per call, but I'm not sure which version this was tested with. Under version 1.6.1820, I don't think there's a limit on the number of choice *calls* in a single result, but only 20 choices can be displayed on screen at one time: if more are displayed, the game will freeze when the player clicks one (this applies to scripts as well as dialogue results). More than 20 *possible* choices won't cause problems as long as no more than 20 are actually displayed (i.e. using conditional statements to select 20 or less from a larger number of choices is OK).

On using Choice in script: I don't advise using it in a script that also contains a StartScript command, as this can also freeze the game sometimes. If you need to use both, you might try delaying the StartScript part until out of menumode if possible or using some condition to ensure that the choices can't be given more than once (not tested).

## Adding to the journal and testing journal entries

**[no fix] Journal, "Journal\_ID", Index\_enum**

```
Journal, MG_BCShroomsCombat, 10
```

This adds a journal entry to your in-game journal. The journal entry must have previously been set up in the dialogue editor. Index references which part of a journal topic is added. Beware of using simple names for journal topics, adhere to Bethesda's two letter standard (see above example) – otherwise the journal entry might show up as a regular conversation response, just like any other, if the topic title shows up in a conversation!



**[no fix] SetJournalIndex "Journal\_ID" index\_enum**

```
SetJournalIndex "MG_BCShroomsCombat" 99
```

SetJournalIndex will set the index to the specified value, whether or not an entry exists for that index. This can be used to restart quests or repeat sections (by setting the index backwards), or for temporary storage of simple flags that don't need an own journal entry. However, it should not be used to finish quests, as the quest will not be removed from the active quests list (Journal does not have this problem, so it is to be preferred for finishing quests).

**Note:** If the index set does not have a valid journal entry (i.e. that index isn't defined in the "info" section of the dialogue window), the value will be lost when a savegame is reloaded (the index will revert to the last valid entry given). Therefore this can also be used to detect if the player has reloaded the game:

```
if ( ( getjournalindex "dummy" ) != 100 )
    Messagebox "You just reloaded, Cheater!!!"
    setjournalindex "dummy" 100
endif
```

Where "dummy" is any journal topic that has no text for index 100.

The best thing is that it's most easy to use this in dialogue: Send the player to the "test of courage", and set the journal index in dialogue-result. When the player comes back, and the index differs, then the player has failed the test. (Info on this function provided by JOG).

**[no fix] ClearInfoActor**

This is a function that is used in the results section of a dialogue response – using this function will stop the response from appearing in the PC's journal (under "Topics"). Useful to avoid cluttering the topic with useless information. Note however that if all responses for a topic use ClearInfoActor, the topic will still appear in the topics list in the journal, but clicking it will take the player to a blank page.

**[no fix] GetJournalIndex, "JournalID" (returns short)**

```
If ( GetJournalIndex, MG_BCShroomsCombat == 10 )
```

This function returns the index of the current journal entry for that journal topic (meaning the last index given, or the last index given for which an entry exists in the case of SetJournalIndex and a reloaded savegame). This is very convenient for keeping track of quest

advancement, and to have a script react according to what parts of the quest have already been performed.

**Sample Script:** Here is a short script demonstrating the use of both functions from the game:

```
Begin attack_slave
short nolore
If ( GetDeadCount "Vorar Helas" > 0 )
    return
endif
if ( GetJournalIndex "MV_SlaveMule" < 102 ); if PC has not yet reached a certain point
    If ( GetDistance, "Rabinna" < 512 )
        Rabinna->AiWander 0 0 0 0 0 0
        StartCombat, "Rabinna"
        Journal "MV_SlaveMule", 102 ; add journal entry
    endif
endif
End
```

## Special dialogue-only functions

Among the functions available for defining dialogue conditions in the dialogue editor there are a few that do not have a direct equivalent in scripting functions. By using dialogue (e.g. using ForceGreeting, voice-dialogue or strategically placed quest NPCs) and the dialogue result field, you can still make use of these functions for scripting purposes, e.g. by setting a global variable in the result field. Examples of such functions are:

### PC Sex (dialog)

This is 0 if the player is male and 1 if the player is female.

**Note:** This is the only known way to determine the player's sex. You could use this to set a global variable that contains the Player's gender - the most inconspicuous way would be to do this via voice dialogue, e.g. one could put a silent hello greeting for all NPC's on top of the list that sets a global variable "PC\_sex" to 1 when the player is female, 2 when the player is male. The topic should be filtered to only be active when PC\_sex equals 0.

### Talked to PC (dialog)

This is 1 if the speaker has talked to the player and 0 otherwise. You can use this to have someone say something the first time you speak with them – or for our scripting purposes to mark this person as "known" by the player.

**Note:** The Talked to PC flag is subject to the "72-hours bug", i.e. it will by default be reset if the NPC's cell has not been active for 72 game hours or longer. See *ForceGreeting* for specific workarounds, and the "72-hours bug" section under "Tips and Tricks" for more general information.

### Rank Requirement (dialog)

Rank Requirement checks to see if you "qualify" for the next rank in the speaker's faction.

- Returns 0 if you do not have enough Faction Reputation and do not meet the skill requirements.
- Returns 1 if you meet the skill requirements, but do not have the Faction Reputation.
- Returns 2 if you have the Faction Reputation, but do not meet the skill requirements.
- Returns 3 if you qualify.

### PC Clothing Modifier (dialog)

This is the total value of all the clothing and armor the player is wearing. The value of your equipment changes the disposition of people in the game.

### Friend Hit (dialogue)

Used in dialogue for when you attack a member of your group (like a follower)

The return values are:

0 = never been hit

1 = hit by pc 1st time

2 = hit by pc 2nd time

3 = hit by pc 3rd time

4 = hit by pc 4th time and the npc/creature is not in combat with the PC

Friend hit will reset if the player hits a different actor. It also appears to reset if the actor starts combat, even if there are no hits (I tried "ActorID->StartCombat ActorID" - i.e. "start combat with yourself" - in dialogue results, and this seemed to reset friend hit).

### Changing the Hello setting

#### Get/Mod/SetHello

Changing this changes it for ALL references of the Actor. Some info from the helpfile: Hello equates to the distance at which the Actor will stop, face the PC and say hello. The setting (which defaults to 30) is multiplied by the game setting, iGreetDistanceMultiplier, which defaults to 7. Thus, a setting of 30 yields a hello distance of 210 (just under 10 feet).

### Useful dialogue variables

A number of short variables are used by Bethesda to block certain dialogue. These must simply be declared in a script on the actor, no value is required.

They are checked for using the Not Local filter as described in the helpfile:

|         |                                       |
|---------|---------------------------------------|
| Nolore  | Blocks most non-specific dialogue     |
| NoIdle  | Blocks Idle voice, used for vampires  |
| NoFlee  | Blocks flee voice, used for vampires  |
| NoHello | Blocks Hello voice, used for vampires |

This is true if the speaker does not have this local variable. Unlike most "Not" functions, this one does care what you set the variable to. Both the dialogue and the variable itself should be set to 0. This can be confusing. Here is a table of how this works:

| Not Local<br>(in dialogue) | Variable Exists<br>(y/n) | Value<br>(in the script) | Pass?<br>(speaker will say this) |
|----------------------------|--------------------------|--------------------------|----------------------------------|
| = 0                        | No                       | NA                       | Yes                              |
| = 0                        | Yes                      | 0                        | No                               |
| = 0                        | Yes                      | 5                        | Yes                              |
| = 1                        | No                       | NA                       | Yes                              |
| = -3                       | Yes                      | -3                       | No                               |

## Changing and testing Skills, Attributes, and other Stats

### *Get, Set, and Modify stats - general remarks*

**GetStat** (returns float)

**SetStat, var\_float**

**ModStat, var\_float**

```
Set floatvar to ( Player->GetHealth )
Player->SetWillpower, 20
Player->ModHealth, floatvar
```

This is really a whole family of functions that can alter player and Actor stats, AI settings and more. Replace *Stat* with any of the game stats, attributes, AI-settings, resistances, reputation, etc. (**list see Appendix**). Positive values are added to current stat, negative values subtracted. GetStat returns a float value with the **current** value of *Stat* (Not the maximum or base "natural" value of that stat for the player, but what is currently used by the game, e.g. it could be boosted by magic or reduced by disease).

SetStat sets the stat's **base and current** value to the given value.

ModStat adds (positive values are added to current stat, negative values subtracted) the given value to both the **base and current** value of *Stat*. ModStat can not set an attribute beyond its natural limit (100) while SetStat can. Presumably the behavior is equivalent for other *Stat*'s.

**Note:** This is not true of some of the more unusual stats like resistances, which can be negative (weakness) and aren't limited to 100 either.

There are so many things you can do with this set of functions that it is not very useful to provide a sample script. Take a look at the Marksman Toggle script in the Tips and Tricks section for a good example. The script given under "Resurrecting a dead Actor" below also uses ModHealth, as do many others.

These commands have a wealth of applications. They could be used for special items, curses, blessings, to gain information on the player's strengths and weaknesses, and to change AI settings (making an Actor more aggressive after player has insulted him, making an NPC uncommunicative at night etc.). Another popular use is to change weapon stats to make NPC's switch their equipment.

**In the 8<sup>th</sup> edition of this guide many of these functions have been sorted into the appropriate chapters (e.g. magic, combat, etc. )**

## ***Determining and changing Actor and player stats:***

### **Determining and changing Attributes:**

***Get/Mod/SetStrength***  
***Get/Mod/SetIntelligence***  
***Get/Mod/SetWillpower***  
***Get/Mod/SetAgility***  
***Get/Mod/SetSpeed***  
***Get/Mod/SetEndurance***  
***Get/Mod/SetPersonality***  
***Get/Mod/SetLuck***

### **Determining and changing Health, Magicka, Fatigue:**

***Get/Mod/SetHealth***  
***Get/Mod/SetMagicka***  
***Get/Mod/SetFatigue***

These return, change or set the vital functions of the PC. For NPC and the player the Get functions will report the current health/magicka/fatigue. GetHealth also works on weapons / armor, but only returns the maximum health. No function is known that reports current item health (Forum info/Mana User).

### **Special use with ModStat only:**

***ModCurrentHealth, var\_float***  
***ModCurrentMagicka, var\_float***  
***ModCurrentFatigue, var\_float***

While ModHealth changes both the maximum and the current health of an Actor for the same amount (e.g. even a healthy Actor would be affected), ModCurrentHealth affects only the current health and can not set health above the original maximum health value for that Actor (so doing ModCurrentHealth, 10000 to an Actor with 70 Health and a current health of 35 would set Health to 70 – Doing ModHealth, 10000 would set him to 10035 health).

### ***Changing Health:***

SetHealth affects the object, not just a reference. If you use SetHealth from a global script on a non-unique ID, it will apply to all instances of the object that the player has not encountered yet (but if the player has already encountered that reference, it won't be affected). If SetHealth is used from a targeted script on a non-unique ID, it will have much the same effect as a global script but will also affect the script's target, even if the player has encountered it already. ModCurrentHealth is the reverse: it affects the reference rather than the object. SetHealth in the console window with a reference selected will also affect only that reference. Note that the same does not apply to other derived attributes: If you place a new reference, Fatigue will be at editor level, and Magicka will be auto-calculated from Intelligence (regardless of editor level).

***GetHealthGetRatio*** ***(returns float)***

This function returns the health ratio of the Actor as a float value from 0 to 1, e.g. 1 means 100% health, 0.9 means 90% health and 0 means, well, dead I guess. This replaces the erroneously listed function GetHealthRatio listed in the original helpfile.

If you want to know an Actors maximum health (Remember, GetHealth returns your *current* health points) you can use this:

```

Float MaxHealth
Float CurrentHealthRatio

Set CurrentHealthRatio to ( "Actor ID"->GetHealthGetRatio )
if ( CurrentHealthRatio > 0 )
    Set MaxHealth to ( ("Actor ID"->GetHealth ) / CurrentHealthRatio )
else
    Set MaxHealth to 0
endif

```

## Determining and changing Skills:

Changing weapon skills can be used to change what weapon the NPC uses by default. This doesn't appear to work correctly for armor – the NPC will only wear armor based on the skills set in the TES CS. If you know the item ID, and have either Tribunal or Bloodmoon, you can use the Equip function to force the NPC to use it (forum Info/Vorwoda\_the\_Black). See the tips and tricks section for an example.

What's not obvious is the range of acceptable values for a skill: its not just 0-100 as you might expect. In fact skills appear to be stored as a float so you can set some large numbers in there, but there are some checks: you can't set negative values, and decimal points are discarded when saving/loading (Thanks FreshFish).

```

Get/Mod/SetBlock
Get/Mod/SetArmorer
Get/Mod/SetMediumArmor
Get/Mod/SetHeavyArmor
Get/Mod/SetBluntWeapon
Get/Mod/SetLongBlade
Get/Mod/SetAxe
Get/Mod/SetSpear
Get/Mod/SetAthletics
Get/Mod/SetEnchant
Get/Mod/SetDestruction
Get/Mod/SetAlteration
Get/Mod/SetIllusion
Get/Mod/SetConjuration
Get/Mod/SetMysticism
Get/Mod/SetRestoration
Get/Mod/SetAlchemy
Get/Mod/SetUnarmored
Get/Mod/SetSecurity
Get/Mod/SetSneak
Get/Mod/SetAcrobatics
Get/Mod/SetLightArmor
Get/Mod/SetShortBlade
Get/Mod/SetMarksman
Get/Mod/SetMercantile
Get/Mod/SetSpeechcraft
Get/Mod/SetHandToHand

```

## Determining and changing Level

**Get/Mod/SetLevel** (only works with Set and Get)

Sets the Actors level, and only that. Skills and stats are not automatically increased, nor does the levelling menu come up when you call this for the player.

Note that SetLevel does not accept variables (using variables will not generate an error, but the actor's level will be set to 0).

## **GetStat, ModStat and SetStat: A concerned modder's guide. - Galsiah**

*Throughout the term “base stat” means the value of the stat when yellow – i.e. unmodified by any in game bonus / penalty apart from fortification abilities.*

### **GetStat:**

Always returns the current stat – including any bonus / penalty. No nasty side effects as far as I'm aware, so GetStat should always be safe. There is (sadly) no GetBaseStat function. MWSE has these, but they don't include e.g. racial fortification abilities in the base, and usually you'd want them included.

Most of the problems of ModStat and SetStat explained below are only really important for the player (and possibly companions). These functions won't do anything worse than screwing up the stats they operate on, so for normal NPCs, there's not much point worrying about all this.

As with pretty much anything in Morrowind modding, if you have a choice to use scripting or something else, then use something else. For ModStat / SetStat, the “something else” will usually be fortify/drain stat spell effects or curses. For most purposes, standard effects will work as well as ModStat and SetStat, and will be much less buggy.

### **ModStat:**

Usually increases (or decreases) the stat concerned by the amount given. Both the current stat and the base stat are affected. Usually preserves the amount of fortification or damage on the stat.

E.g. for a player with strength 50(base) + 10(fortification) = 60(current)  
Player->ModStrength 10 will give 60(b.) + 10(f.) = 70(c.), just as you'd expect.

Limitations: ModStat can never decrease a stat below zero. It also cannot increase a base stat to a value over 100. Trying to Modstat below zero or above 100 can cause trouble in the following ways:

Strength = 30(b.) + 0(f.) = 30(c.)

ModStrength, -50 gives

Strength = 0(b.) + 0(f.) = 0(c.)

ModStrength 50 [hoping to undo the first modstrength]

Strength = 50(b.) + 0(f.) = 50(c.) – and the player has a permanent bonus.

You can try to avoid this by instead checking that you don't reduce the stat by more than the current value, but that won't always work. For example:

Strength = 40(b.) + 10(f.) = 50(c.)

ModStrength -50

Strength = 0(b.) + 0(f.) = 0(c.)

ModStrength 50

Strength = 50(b.) + 0(f.) = 50(c.) – and the player's base strength is increased.

If he then removes the fortification, his strength will show up as damaged. Restoring and replacing the fortification will leave him with:

Strength = 50(b.) + 10(f.) = 60(c.)

It's never safe to ModStat down by more than the player's base stat. Given that there's no failsafe method to determine the player's base stat, this is annoying [the only ways I know to determine the player's base stat are the method which I use in GCD – complicated, and doesn't work correctly when the player's stat is damaged -, using MWSE, which I think doesn't include permanent abilities in the base (for most purposes you'd want permanent

abilities to count towards the base). Even systematically removing every conceivable bonus / penalty – which is a drag anyway – won't always work with other scripted mods].

Going over 100 has similar problems. The following is fine:

$$\text{Strength} = 70(\text{b.}) + 20(\text{f.}) = 90(\text{c.})$$

ModStrength, 20 gives

$$\text{Strength} = 90(\text{b.}) + 20(\text{f.}) = 110(\text{c.})$$

ModStrength -20 gives

$$\text{Strength} = 70(\text{b.}) + 20(\text{f.}) = 90(\text{c.}) - \text{all fine.}$$

However, this also happens for some reason:

$$\text{Strength} = 70(\text{b.}) + 20(\text{f.}) = 90(\text{c.})$$

ModStrength, 50 gives

$$\text{Strength} = 100(\text{b.}) + 40(\text{f.}) = 140(\text{c.}) - \text{already screwed up.}$$

ModStrength -50 gives

$$\text{Strength} = 50(\text{b.}) + 40(\text{f.}) = 90(\text{c.}) - \text{further screwed up.}$$

This problem arises because if a stat is fortified or damaged, and the base is not 100, ModStat always increases the current stat by the value you give it, even if the base stops at 100. If the base is 100, ModStating won't have any effect. If the stat is equal to its base value, ModStat will behave normally.

So this is fine (so long as you don't ModStat -50 afterwards):

$$\text{Strength} = 70(\text{b.}) + 0(\text{f.}) = 70(\text{c.})$$

ModStrength, 50 gives

$$\text{Strength} = 100(\text{b.}) + 0(\text{f.}) = 100(\text{c.})$$

And this is fine:

$$\text{Strength} = 100(\text{b.}) + 20(\text{f.}) = 120(\text{c.})$$

ModStrength, 50 gives

$$\text{Strength} = 100(\text{b.}) + 20(\text{f.}) = 120(\text{c.})$$

But this isn't:

$$\text{Strength} = 99(\text{b.}) + 20(\text{f.}) = 119(\text{c.})$$

ModStrength, 50 gives

$$\text{Strength} = 100(\text{b.}) + 69(\text{f.}) = 169(\text{c.}) - \text{oh dear.}$$

Of course as a modder you won't know in general what the Base + Fortification is before you use the ModStat function, so you have no way to compensate for errors even once you know what can go wrong. Joy!

Ok, so as long as you never try to ModStat the current stat over 100, everything should be fine, right?

Sadly not:

$$\text{Strength} = 95(\text{b.}) - 10(\text{damage}) = 85(\text{c.})$$

ModStrength, 15 gives

$$\text{Strength} = 100(\text{b.}) + 0(\text{f.}) = 100(\text{c.}) - \text{OK so far}$$

ModStrength, -15 gives

$$\text{Strength} = 85(\text{b.}) + 0(\text{f.}) = 85(\text{c.}) - \text{Permanent strength damage.}$$

So under what circumstances will ModStating up or down give reliably predictable results?

Only when you know the base value of the stat.

### **Can you reliably work out the base value of the stat?**

No – only in some situations is it possible (the process is explained below, and implemented in my Gals\_Sk\_Acrobatics script in GCD). Even then it's not easy. (it's worth checking script extenders for updates though)

### **SetStat:**

Sets both the base and the current value to the value you give it (also accepts local variables). Can set base and current to values over 100. Accepts negative values, which can be useful in conjunction with ModStat (see below).

This will pretty much always cause problems if the player's current stat is not equal to their base stat. If their stat is fortified, and you SetStat it, it'll turn yellow at the value you give it. Removing the fortification, then restoring will give the player a permanent bonus. If their stat is damaged, SetStating it will again turn it yellow, but this time at a lower value than you (probably) intended. They will instantly have their base knocked down to the value you set.

Using SetStat is therefore never even slightly safe unless you know the player's base stat, and compensate accordingly. While you can't guarantee that ModStat won't cause trouble, you can almost guarantee that SetStat will.

Using SetStat is therefore almost always a bad decision – if you're ever not sure whether it's a bad decision, then it is.

### **Then are these functions useless!?**

Not always. You can use them on NPCs without worrying too much. Using them on companions could occasionally not have the effect you want, but it's unlikely the player would notice.

Using ModStat on the player can be safe, so long as you're careful – e.g. to increase strength by (up to) 10 points temporarily, you could:

Give the player a very strong restore strength curse for a frame or two. (you can then be sure his strength isn't damaged)

ModStrength by MIN{ 10, 100-current }

...

ModStrength down by the same amount when you want the effect to finish.

You can never be sure that giving a temporary penalty won't cause trouble, but if you only reduce the stat by at most 30, then it'll usually be fine since most players start with all stats that high. Restoring the stat after you've reduced it like this could cause trouble if the player has since gained base stat points, and his stat started close to 100.

### **Some "cunning" tricks with SetStat / ModStat:**

Warning: using the following tricks might cause even more trouble than using the functions normally. Use with care. Conflicts are likely.

### **Finding the base of a stat:**

You can find the base of a stat using the (mis)behaviour of ModStat, as follows:

- (1) Make sure that the player's stat isn't damaged [You have been tracking increases in the natural values of the stat since the beginning of the game, haven't you? If you haven't, then you have no way of knowing if it's damaged – you could check the base value on installation of your mod, so long as you ask the player to install when stats aren't damaged / fortified].
- (2) If it is damaged, give up. (you might want to check script extenders)
- (3) If it isn't:
- (4) Store the current value of the stat.
- (5) Mod the stat up to 100 if it isn't already 100 or more.
- (6) ModStat, 1 as many times as you can while the stat still increases.
- (7) The fortified part of the stat is the value it reached minus 100.
- (8) The base part of the stat is the current value minus the fortified part.
- (9) Return the stat to its current value (DON'T use SetStat, use ModStat).

Armed with the base value, you can now use ModStat and SetStat wisely and safely, so long as you're very careful.

### **Damaging a stat:**

To set a stat to e.g. 50 damaged from 70, you can do the following:

Player->SetStat, -20 [Base and current are now both -20]

Player->ModStat, 0 [Necessary: sets the base to 0]

Player->ModStat, 70 [Base = 70, current = 50 (red)]

This level of precision is not possible in general using e.g. damage stat curse effects, but it's not usually necessary either. The above can also be done within one frame, whereas a curse effect might take a second to kick in. Again, the speed can be useful, but is usually unnecessary.

### **Fortifying a stat:**

To set a stat to e.g. 120 fortified from 90, you can do the following:

Player->SetStat 130 [Base and current are now both 130]

Player->ModStat, 0 [Necessary: sets the base to 100]

Player->ModStat, -10 [Base = 90, current = 120 (white)]

# Combat

## *Initiating and ending combat*

**StartCombat, "ActorID"**

**StopCombat**

```
"Actor_ID1"->StartCombat, "ActorID2"  
"Actor_ID1"->StopCombat
```

StartCombat and StopCombat are used to set an Actor into combat mode or back into normal mode. Start combat will make the calling Actor attack the Actor supplied as the argument. While StopCombat seems to be "safe" to use "every frame", you should supply a do once condition of some sort when issuing the StartCombat command, otherwise the Actor might not do anything. Nevertheless continuous StopCombat is very dangerous to use, because it makes the NPC completely helpless: it will not retaliate when attacked (which however allows you to create a real pacifist...). Another caution regarding StopCombat is that it stops combat for **all** actors involved, other than the player.

Once in combat mode the AI settings of the Actor apply normally, e.g. if the Actor has a high flee setting, he will flee despite the StartCombat command. For this reason, you will often see that the Fight rating is also changed when initiating combat in many scripts:

```
If ( GetDeadCount, "My Friend" > 0 )  
    StartCombat, Player  
    SetFight, 100  
endif
```

The same applies to StopCombat: if the actor has a high Fight rating he will go briefly into an idle stance (sheathe weapon or un-ready spell), then quickly enter combat with the player again; to stop combat completely, set the actor's Fight rating to a low value as well.

## *Detecting Attack*

**[no fix] OnPCHitMe (is local short variable)**

```
Short OnPCHitMe  
If ( OnPCHitme == 1 )
```

A local game variable (not a function, you must declare it as a variable as shown above) that gets set to 1 when the player hits the calling Actor. Must be manually reset. It seems the use of the variable "short-circuits" normal NPC behavior in that an NPC with a script that uses this variable will not attack on its own accord. If you don't want the Actor to remain passive you have to manually *StartCombat* (see example below). Once the Actor is in combat mode, *OnPCHitMe* does not report any further hits by the PC. Except, according to information on the forum, *OnPCHitMe* gets reset (to 0) if another Actor hits the calling Actor after the PC did, then the variable gets reset to 1 if the player hits again.

**Note:** According to info provided by Nigedo, OnPCHitMe also registers if the PC commits a crime, and the Actor has a sufficiently high alarm setting.

**Example Script:** An example from my traveling merchants mod, to make a guar handler defend his charge while not in AIFollow mode, I attached this to the guar:

```
Begin _HB_Adros_GuarDefend  
  
float timer
```

```

short attackstate
short OnPCHitMe

if ( OnPCHitMe == 1 )
    set attackstate to 1
    set OnPCHitMe to 0
endif

if (attackstate == 1)
    StartCombat, player
    set timer to ( timer + GetSecondsPassed)
    if ( timer >= 1 )
        set timer to 0
        if ( GetLOS, HB_adros_darani == 1)
            HB_adros_darani->StartCombat, Player
            set attackstate to 0
        endif
    endif
endif
End

```

### GetAttacked (returns Boolean/short)

If ( Actor->GetAttacked == 1 )

Returns 0 if the Actor has never been attacked and 1 if he has ever been attacked.

#### Example script: Uupse protects Yagrum Bagarn:

```

Begin uupse_Bagrum

short doOnce

if ( doOnce == 0 )
    if ( "yagrum bagarn"->GetAttacked == 1 )
        StartCombat player
        SetFight 90
        SetDisposition 0
        set doOnce to 1
    endif
endif

end uupse_Bagrum

```

### GetTarget, "Actor ID" (returns Boolean/short)

If ( Actor->GetTarget, Player == 1 )

GetTarget tests to see if the target is in focus. For the player, it's the test to see if the target is currently in the crosshair and can be activated. Of course NPCs don't have a crosshair like the player, but they get the same logic. (Thanks to cdcooley for the explanation)

If you want to use GetTarget to see if one actor is fighting another, you would have to combine it with GetWeaponDrawn and GetSpellReadied.

```

short temp
set temp to GetWeaponDrawn
set temp to ( temp + GetSpellReadied )
if ( temp > 0 )
    if ( GetTarget, Player == 1 )
        ; In combat with the player here
    endif
endif
endif

```

**HitOnMe, "Weapon ID" (returns Boolean/short)**  
**HitAttemptOnMe, "Weapon ID" (returns Boolean/short)**

These functions return true (1) for 1 frame if the calling Actor is successfully hit or if it was attempted to hit it with a specified weapon. HitOnMe is used only in the LorkhanHeart script (only look at that if you have finished the game or don't mind severe spoilers). I guess it could be a nice function to script any kind of fight of the "you need this special weapon to kill this particular monster" type.

## **Combat related Get/Mod/Set AI functions: Fight, Flee, Alarm**

### **Get/Mod/SetFight**

Some info from the helpfile:

An Actor's fight setting determines how prone the Actor is to attacking the PC. Mod/Set Fight appear to also affect all new references of the actor you may add after calling this function (i.e. it affects the object, not just a reference). When an Actor's fight setting hits 100, they will attack the PC.

Player actions will increase (or decrease) an Actor's fight setting. These are:

| PC Action      | Default Value                | Game Setting Formula                                  |
|----------------|------------------------------|-------------------------------------------------------|
| PC Distance    | 20 - (Char Distance * 0.005) | iFightDistanceBase - (Char Distance * fFightDistMult) |
| Attack Actor   | 100                          | iFightAttack                                          |
| Disposition    | ( 50 - Disposition ) * 1     | (50 - Disposition) * fFightDispMult                   |
| Stealing       | 5 * Item Value               | fAlarmStealing * Item Value                           |
| Pick Pocketing | 25                           | iAlarmPickPocket                                      |
| Trespassing    | 25                           | iAlarmTresspass                                       |
| Taunting       | From Persuasion Formula      |                                                       |
| Intimidation   | From Persuasion Formula      |                                                       |
| Bribery        | From Persuasion Formula      |                                                       |

The following table gives you the resulting general behavior:

|     |                                                                          |
|-----|--------------------------------------------------------------------------|
| 100 | Always Attacks                                                           |
| 95  | Will Attack as PC gets close (3000 units)                                |
| 90  | Will Attack as PC gets close (2000 units)                                |
| 80  | Will Attack as PC gets close or if he dislikes you (1000 units, 40 Disp) |
| 70  | Will Attack if close and strong dislike (1000 units, 35 disp)            |
| 60  | Will Attack if he dislikes you and you get close (Disp below 30)         |
| 50  | Will Attack if he hates you (Disp at 0)                                  |
| 40  | Will attack if he dislikes you, and you get close. (500 Units, Disp 10)  |
| 30  | Will Attack if hates you and you commit crime.                           |
| 20  | Will Attack if dislikes you and multiple crimes.                         |
| 10  | Will attack if he hates you and you do multiple crimes on him.           |
| 0   | Will ONLY attack if attacked first.                                      |

### **Get/Mod/SetFlee**

Changing this changes it for ALL references of the Actor (see note).

Setting this to a higher value will make the Actor more likely to flee, but this may not always be the result, as the Actor will also use other factors like how much damage they can give out,

or other strategies they may use such as magic and ranged combat. The behavior is strongly influenced by a number of GameSettings that are listed below, and a number of mods (e.g. by wakim and maxpublic) have tweaked these values to allow for more realistic fleeing behavior.

### **Get/Mod/SetAlarm**

Changing this changes it for ALL references of the Actor (see note).

Some info from the helpfile: When a crime is committed, and it is detected by an NPC, they will shout something at the player, this also notifies other NPCs in the area.

When the NPCs hear this, they adjust their settings based on their alarm setting. The higher the alarm setting, the angrier they will get.

If an NPC has an alarm of 100, he will put gold on the PC's head if they hear of a crime.

If the NPC with alarm 100 is also of class "Guard", they will have extra behavior:

Intercept the PC, by running up and arresting the PC.

If the PC's CrimeLevel is over 10000, they will attack on sight, instead of initiating dialogue.

Guards will also attack any creatures they can see that are attacking people (including the PC). If the player has followers (companions or other NPCs in AIFollow mode), the presence of a guard-class NPC in the party may also cause the non-guard followers to attack hostile creatures before blows have been exchanged (normally a non-guard follower would do nothing until a hit has occurred). -(Forum info/Neko)

**Note:** When you use these functions to alter the settings for an actor, it alters the current reference of the actor AND the definition of the actor. What this means is if you encounter a new actor of that id that you haven't yet met, he will have the new alarm/Fight setting. Also, if you leave the cell where an actor still has the old value, rest for 3 days (to disconnect them from memory) then re-enter the cell, he will take his value from the definition of the actor i.e. the new alarm setting (Forum info / Cortex).

## ***Keeping track of kills and knockouts***

### **OnDeath (returns Boolean/short)**

```
If ( Actor->OnDeath == 1 )
```

Returns 1 for 1 frame when the Actor is killed. OnDeath seems to reset itself once it is used. This also means that only one script can reliably detect death this way: if you have both a global and a local script using this function, only the global script will detect OnDeath. An alternative would be to use the GetHealth function. In the following script only the first message box will be displayed (forum info / Argent, ThePal):

```
begin personScript
if ( OnDeath )
messagebox "1"
endif

if ( OnDeath )
messagebox "2"
endif

end
```

### **OnMurder (returns Boolean/short)**

```
If ( Actor->OnMurder == 1 )
```

Returns 1 for 1 frame when the Actor is murdered. The conditions for *OnMurder* are not entirely clear to me – from the context of its use in the game however, it seems that *OnMurder* gets set when you are reported as a murderer to the law ("your crime has been reported"). So a murder only happens when you kill someone illegally AND are seen.

**Sample Script:** this sets a variable that is used in the "Redoran Hortator" dialogue topic to determine if the player has killed a councilor:

```
begin RedoranCouncilor

;no lore...
short noLore

;for HT_Monopoly
short mageMonopolyVote

;for Hortator dialogue...
if ( OnDeath == 1 )
    if ( OnMurder == 1 )
        Set RedoranMurdered to 2
    else
        Set RedoranMurdered to 1
    endif
endif
End
```

#### **OnKnockout (returns Boolean/short)**

```
If ( Actor->OnKnockout == 1 )
```

Returns true for one frame when the Actor is knocked unconscious (e.g. in hand-to-hand combat)

#### **[no fix] GetDeadCount, "Actor ID" (returns short)**

```
If ( GetDeadCount "divayth fyr" > 0 )
```

The function returns the number of references (individuals) of type "Actor ID" that have been killed. A useful function for quest scripting to keep track of which NPCs are still alive. Note that there is an equivalent function for dialogue as well. Other uses are imaginable, e.g. building a reputation with certain monsters that might flee you instead of fighting after you killed more than 100 of them, etc.

#### **Sample Script:**

GetDeadCount is often used to check if a certain NPC is dead. It is advisable to use "> 0" in such cases, as you never know if another mod might add another instance of that ID, so it's better to play it safe.

```
Begin araraUvulasScript

short noLore

if ( CellChanged == 0 )
    return
endif

if ( GetDeadCount "Neloth" > 0 )
    Disable
endif

End
```

## ***Resurrecting a dead Actor***

### **Resurrect**

gateway\_haunt->Resurrect

This function brings an Actor back to life. His stats and inventory will be reset, basically he "respawns". There is a bug when you use this function on the player – it will stop the PC (and all Actors) from casting magic. After saving and reloading this side effect goes away.

**Note:** The Puzzle Canal Script shows an alternative: it simply uses GetHealth <10 to determine when player is "nearly" dead and then "resurrects" him by giving him his health back – so the player actually never really dies.

**Sample Script:** some people are just tougher than others...

```
Begin dandrasScript

short deathbed
float dandrasHealth

if ( deathbed == -1 )
    return
endif

set dandrasHealth to GetHealth

if ( dandrasHealth <= 50 )
    if ( dandrasHealth < 1 )
        Resurrect
        ModHealth 100
    endif
    set deathbed to 1
endif

if ( deathbed == 1 )
    ForceGreeting
endif

End dandrasScript
```

## Crime

### ***Determining and changing Crime Level***

***Get/Mod/SetPCCrimeLevel (PC Only)***

*PCCrimeLevel* governs the gold you have to pay to be cleaned of crimes, influences NPC disposition and how guards react to you. See also the PayFine function.

### ***Jailing the PC***

***[no fix?] GotoJail***

Sends the PC to the (closest available) prison, more exactly speaking to a PrisonMarker (Door object) and applies the usual prison penalties.

#### **SampleScript:**

Here is a cool little scripted item by B from the Modern Adventurer mod. The cursed Holiday Pants that send you to prison:

```
Begin Holiday_script

Short OnPCEquip
Short message

if ( OnPCEquip == 1 )
    if ( MenuMode==1 )
        return
    else
        Set message to Random 2
        if ( message==0 )
            MessageBox "The holiday pants contain a mighty enchantment of happiness.
Lots of happiness. When the guards found you doing the Can-Can on top of the nearest silt
strider port, they were not very amused.", "ok"
        elseif ( message==1 )
            MessageBox "The holiday pants make you scream and shout with joy as you
relive the happiest days of your childhood. The guard that brings you back to your senses is
in stark contrast to this experience", "ok"
        endif
        Player->GoToJail
    endif
Set OnPCEquip to 0
endif

End holiday_script
```

### ***Clearing the PC of crime***

***[no fix] PayFine***

The PayFine function removes the stolen items from the PCs inventory; it does not remove any gold. Call after paying a crime fee to clean AI. Also puts the PCs hands down (that is not ready to cast or fight).

***[no fix] PayFineThief***

Like PayFine function but *does not* remove stolen items from the PCs inventory. Call to "clean AI". May have incorrectly removed stolen items before one of the patches. For examples check below, under "Useful global variables"

### ***Detecting crime***

**[no fix] GetPCCrimeLevel (returns short)**

Reports the current crime level of the PC. Can be used to detect whether a crime the PC has committed has been seen. See the "Bill\_MT\_writxxxxx" scripts for examples of its use.

An alternative was reported by Nigedo:

#### **OnPCHitMe**

If you declare OnPCHitMe in an NPC's script, any crime that they are aware of causes this function/variable to return True. The crime does not actually have to be committed against that NPC, they just have to have a high enough Alarm setting to care about a crime being committed within range, and the crime will count as a melee hit on them of zero damage. Although this makes OnPCHitMe less reliable for detecting just attacks on the NPC it is declared on (I had to use a different method for the script I was actually working on), it is potentially useful for detecting crimes taking place.

It is possible to use this to detect all crime events in one script, without needing an NPC to "report" them, i.e. increase Player's bounty, or needing to check or adjust PCCrimeLevel.

I found that the following alarm settings will (usually) cause OnPCHitMe to return True for the these events:-

| <b>Event</b>      | <b>Minimum Alarm</b> |
|-------------------|----------------------|
| Any theft         | 10                   |
| Assault of an NPC | 90                   |
| Murder of an NPC  | 10                   |



**[no fix] GetPCInJail (returns Boolean/short)**

Bloodmoon adds a function that can be checked to see if the PC is in Jail. The function will return 1 if traveling/in jail, zero otherwise. This is used in the werewolf change script to stop the PC from changing if either of these states are the case.

#### **Sample script:**

```
if ( PCWerewolf != 1 ) ; DON' RUN IF PLAYER ISN'T WEREWOLF
    return
endif

if ( GetPCinJail == 1 )
    return
endif

if ( GetPCTraveling == 1 )
    return
endif
```

## ***Useful global variables***

### **CrimeGoldDiscount (is global short)**

Contains the amount of gold needed to pay the reduced fine at the thieves guild.

### **CrimeGoldTurnIn (is global short)**

Contains the reduced fee you have to pay when you turn yourself in.

### **PCHasCrimeGold (is global short)**

Used in dialogue conditions. Gets set to 1 if player has enough gold to pay for his crimes.

### **PCHasGoldDiscount (is global short)**

Used in dialogue conditions. Gets set to 1 if player has enough gold to pay thieves guild discount on the "price on your head".

### **PCHasGoldTurnIn (is global short)**

Used in dialogue conditions. Gets set to 1 if player has enough gold to the reduced crime fee that is charged when turning yourself in.

**Example:** A **dialogue result field** for the topic "price on your head", for paying a fine at the thieves guild:

```
Player->RemoveItem Gold_001 CrimeGoldDiscount  
SetPCCrimeLevel 0  
PayFineThief
```

For comparison, this is the result fields that guards use when you have a price on your head and turn yourself in (you find this under Greeting 0):

```
Player->RemoveItem Gold_001 CrimeGoldTurnIn  
SetPCCrimeLevel 0  
PayFine
```

Or if you get caught and have to pay the normal "fines and compensation":

```
Player->RemoveItem Gold_001 GetPCCrimeLevel  
SetPCCrimeLevel 0  
PayFine
```

# Magic

## *Limiting the use of teleport*

[no fix] DisableTeleporting  
[no fix] EnableTeleporting

Rather self explanatory, these functions turn the ability to use teleporting magic on or off. Nice to keep those magic user types from wimping out of your dungeon ☺. In the original game it's only used when the player encounters Dagoth Ur.

I won't show the whole script as it would be quite a spoiler, but here is the part that uses the function:

```
short teleportDisabled  
if ( teleportDisabled == 0 )  
    DisableTeleporting  
    Set teleportDisabled to 1  
endif
```

This is later reset in the EndGame script.

**Note:** when the original Tribunal is installed this function is effectively **broken**: One of the start-up scripts in Tribunal overrides all other teleport commands and forces teleporting on except within one specific area in Mournhold (thanks to Slink and Riiak for the info). Here is the culprit:

```
Begin TribunalMain  
;check for teleporting  
if ( GetPCCell "Sotha Sil," == 1 )  
    DisableTeleporting  
else  
    EnableTeleporting  
endif  
  
;check levitate  
if ( GetPCCell "Sotha Sil," == 1 )  
    DisableLevitation  
elseif ( GetPCCell "Mournhold" == 1 )  
    DisableLevitation  
else  
    EnableLevitation ; This is why teleporting was always on outside Mournhold  
endif  
  
end
```

With one of the updates this problem was fixed. The latest version of the script looks like this:

```

Begin TribunalMain

short disablestate
short newstate

;by default, enable teleport and levitate
set newstate to 0

;only need to check cells in interiors
if ( GetInterior )
    if ( GetPCCell "Sotha Sil," == 1 )
        ;disable teleport and levitate here
        set newstate to 1
    elseif ( GetPCCell "Mournhold" == 1 )
        ;disable levitate only here
        set newstate to 2
    endif
endif

;if state should change
if ( disablestate != newstate )
    if ( newstate == 1 )
        DisableTeleporting
        DisableLevitation
    elseif ( newstate == 2 )
        DisableLevitation
    elseif ( newstate == 0 )
        EnableTeleporting
        EnableLevitation
    endif
    set disablestate to newstate
endif

end

```

Note: DisableTeleporting does not disable scripted amulets etc. for teleporting. DinkumThinkum suggested the following workaround which uses GetPCCell to check the player's current location. As long as they're in one of the mod cells, nothing happens. If they're *not* where they're supposed to be, then the script teleports them back into the correct area: back to the initial entry point for the mod, for example. This wouldn't be exactly the same as blocking the teleports, but it should make the area totally inescapable until you've fulfilled the modder's conditions for getting out legitimately.

```

Begin DT_Test_BalmoraTrap

If ( GetPCCell, "Balmora" == 1 )
    Return
Endif

MessageBox "Off to Balmora with you!"
Player->PositionCell, -21278, -17613, 534, 0, "Balmora (-3, -3)"

End DT_Test_BalmoraTrap

```

## Limiting the use of levitation



[no fix] EnableLevitation  
[no fix] DisableLevitation

These functions are used to allow and block Levitation magic effects. When DisableLevitation is called, all existing Levitation effects are canceled. When the player tries to cast a spell with a Levitate effect while Levitation is disabled, a notify message is displayed with the text in the GameSetting sLevitateDisabled. Currently this text reads “Levitation magic does not work here.”

### Sample scripts:

This script is on an object in the room with levitation disabled.

```
Begin clampstone

short turnedoff
short gavemessage

if ( turnedoff == 0 )
    DisableLevitation
    if ( gavemessage == 0 )
        set gavemessage to 1
        MessageBox "A strange stone in the roof of this room prevents levitation here."
    Endif
else
    EnableLevitation
    if ( gavemessage == 1 )
        set gavemessage to 0
        MessageBox "The stone has been disabled. You can now levitate in this room."
    Endif
endif

if ( OnActivate == 1 )
    if ( turnedoff == 0 )
        set turnedoff to 1
    else
        set turnedoff to 0
    endif
endif

end
```

This script is on a door leaving the room.

```
Begin enable_lev_on_exit

if ( OnActivate == 1 )
    MessageBox "You leave the presence of the stone..."
    EnableLevitation
    Activate
endif

end
```

## Checking and managing souls and soulgems

### HasSoulgem, "CreatureID"

```
If ( Actor->HasSoulGem, "golden saint" )
```

This function checks if the player has a soul gem containing the specified soul in his inventory. A little used function that could allow some fun quests and new uses for soulgems.

**Sample:** This is part of the StrongSoulCheck script:

```
if ( Player->HasSoulGem "atronach_storm" > 1 )
    Set counter to ( counter + 2 )
elseif ( Player->HasSoulGem "atronach_storm" > 0 )
    Set counter to ( counter + 1 )
endif
```

### RemoveSoulgem, "CreatureID", number\_enum

```
Actor->RemoveSoulGem, "golden saint", 1
```

Removes a soulgem with the specified soul from the players inventory.

Sample: this is the complementary part from RemoveStrongSoul script to the example above:

```

if ( counter > 0 )
    if ( Player->HasSoulGem "atronach_storm" > 0 )
        Player->RemoveSoulGem "atronach_storm" 1
        Set counter to ( counter - 1 )
    endif
endif

```

Note, the player will not be happy if they get Azura's Star taken away by this. Here's a sample solution:

```

short StarCount ;They could have more than one I guess.

if ( OnActivate )
    if ( Player->HasSoulGem "Golden Saint" > 0 )
        set StarCount to ( Player->GetItemCount "Misc_Soulgem_Azura" )
        Player->RemoveSoulGem "Golden Saint" 1
        if ( ( Player->GetItemCount "Misc_Soulgem_Azura" ) < StarCount )
            Player->AddItem "Misc_Soulgem_Azura" 1
        endif
        Player->AddItem Gold_001, 10000
        MessageBox "Thank You, Come Again."
    else
        MessageBox "You have no Golden Saint souls."
    endif
endif

```

### AddSoulGem "creature ID", "soulgem ID"

```
AddSoulGem "atronach_storm", Misc_Soulgem_Grand
```

AddSoulGem adds a soulgem of the specified type and with the specified soul to the players inventory. You can't add more than one at a time with this (giving it a count won't cause any problems, it's just ignored).

### DropSoulgem, "Creature ID"

```
DropSoulGem "atronach_storm"
```

As far as I can tell, this function is **broken** (crashes the game).

### [no fix] OnPCSoulGemUse (is short variable)

The Object is a soulgem and it has been used in either recharging or item making

The soul gems in the game have the following ID's:

#### Soul Gem ID's:

```

Misc_SoulGem_Azura
Misc_SoulGem_Grand
Misc_SoulGem_Greater
Misc_SoulGem_Common
Misc_SoulGem_Lesser
Misc_SoulGem_Petty

```

This function was not used in the original game.

## ***Adding and removing spells and cursing***

**AddSpell, "SpellID"**

**RemoveSpell, "SpellID"**

```
"Actor_ID"->AddSpell "Absorb Speed"
```

The AddSpell function will add the spell to the calling object. This can mean two things: normal spells are added to the actor's spell list. Curses, diseases etc, however will affect the calling object. The same is true for the RemoveSpell function: Normal spells are removed from the list, curses or diseases are removed as effects.

### **Note:**

When you add a spell to a generic NPC, it is added to all NPCs of that ID - unlike items, which are just added to the one NPC.

### **Some general notes:**

You can not remove racial abilities with the RemoveSpell function (forum info). You can remove birthsign abilities, but if the ability was added by the player's birthsign (i.e. during character generation, not later by script), the ability cannot be re-added afterwards (forum info / Rocket).

(Mainly relevant for companions:) Any type of constant magical effect (e.g. abilities, CE enchanted items, etc) on an NPC is reapplied when the NPC changes cells, and when it gets high enough it "wraps around" and usually starts working in reverse. Much of the time this is not noticeable, but if necessary the problem can be avoided by removing and re-adding the effect on CellChanged. In the case of some effects, e.g. water breathing, the game may remove the effect too many times: in this case SetWaterBreathing can be used (note that it must be reset after 72 game hours). (Forum info / CdCooley).

Abilities may sometimes not work as expected. If an ability is removed by script, it may sometimes reappear after reloading a save, or later in the game. Effects on the character's stats may also reappear or not be removed properly on occasion (forum info / exclusiveor77). The other problem with abilities is that damage abilities do permanent damage (i.e. a damage health ability damages maximum health: curses do not have this problem). This does not seem to be true of drain or fortify effects, where it would make a lot more sense. (Forum info / ManaUser)

Curses do, however, have the problem that they can be removed by "Remove Curse" spells.

On the Remove Curse effect, in my tests it worked but somewhat strangely. It's percentile (like dispel rather than cure paralysis) but the percent seemed to be cumulative or something. For example a 1% remove curse spell never worked (as many times as I tested it) unless I first cast something like a 100% remove curse spell. -(ManaUser)

## Casting spells

### Cast, SpellID, "TargetID"

Object\_ID->Cast, "flame", Player

The Cast function makes the calling object cast the spell "SpellID" on the target "TargetID", and Target will suffer or benefit from the effects normally. The calling object does not need to have the spell in inventory: it will be cast regardless.

The spell's target must be specified, even if the spell is "on self". In this case you can specify the caster or the player as the target: if the spell is "on self" it won't make any difference. If you intend to place the target during the game, you will still need to place a reference in the editor. Cast will prefer a reference in the current cell if there is one available, so there's no need to move a reference from a storage cell: you can just place a new one.

#### Notes:

- It was believed that Cast would only work on the PC. At least with Tribunal (not sure about earlier versions) you can use cast to cast a spell from an activator, or any other object, on an Actor. However, make sure the object you are casting the spell from is not in the inventory of another actor/object, as that will result in a CTD.
- You cannot force an actor to cast a spell on an activator, but you can have them cast on themselves, other actors, or the player.
- You can get non actor objects to cast spells on themselves. Just use the player (or some other object) as a target, but cast an "On Self" spell.
- NPCs will not automatically lose magicka from a scripted cast. If you want the NPC to use up magicka in casting, this also needs to be scripted. -(Forum info/Kateri)
- If the spell's target is not in the current cell, the NPC will cast anyway (this doesn't cause any errors, but it does look a little odd - particularly targeted spells). They will also cast 'through' objects, other NPCs, walls... anything that's in the way.
- It is entirely possible for a targeted spell to miss its target. If this could cause problems, it's possible to use a touch spell instead: it will be cast like a targeted spell (from wherever the NPC is standing at the time) and will always succeed, but it will have the same visual effects as a normal touch spell. The downside of this is that if you want an NPC to cast a touch spell normally (from close up) you need to get them into position some other way.

**Sample Script:** The cast function can be used for traps, as in the following example attached to a Container. Note that there is a do once condition here, so that the effect is not cast continuously on the player.

```
Begin Trap_script
short done
if ( OnActivate == 1 )
    if ( done == 1 ) ;do-once condition
        Activate
        return
    else
        Cast, "flame", Player ;damage to player
        set done to 1
        Activate
    endif
endif
End trap_script
```

The next example script uses the *AddSpell* function:

```
begin Item_Cast

short OnPCEquip
short CurseAdded
float Timer

if ( CurseAdded )
set Timer to ( Timer + GetSecondsPassed )
    if ( Timer >= 25 ) ; after 25 s remove the spell. 25 damage done.
        set Timer to 0
        Player->RemoveSpell "ItemFlame"
        set CurseAdded to 0
    endif
endif

if ( OnPCEquip );when item is equipped
    if ( MenuMode )
        return
    elseif ( CurseAdded == 0 )
        Player->AddSpell "ItemFlame" ; a spell of type curse!
        ;custom spell one point of fire damage / s
        set Timer to 0
        set CurseAdded to 1
    elseif ( CurseAdded ) ; Add the spell only once
        Player->RemoveSpell "ItemFlame"
        set CurseAdded to 0
    endif
endif

end Item_Cast

(script by Patrin, edited)
```

The added spell is a custom-made curse spell doing one point per second flame damage. Note that there is again a do once condition implicit in this script. **Failure to have a do once condition can crash the game! Also, it appears that creatures killed with curse spell effects on them cause all other creatures of that type to have the same curse on them. This can be avoided by using 'RemoveSpell' in an 'OnDeath' section of the script. (Forum Info / Argent)**

## Managing and testing for spells

**GetSpell, "Spell\_ID" (returns Boolean/short)**

```
If ( Player->GetSpell, "heal companion" == 1 )
```

Returns true if object has Spell\_ID in inventory. However, this does not work for abilities or powers associated with race or birthsign. Sample script see below.

**GetSpellEffects, "Spell\_ID" (returns Boolean/short)**

```
if ( Player->GetSpellEffects, "flame" == 1 )
```

Returns true if Spell\_ID is affecting calling object. The following could be added to the "trap\_script" discussed under "casting spells" above:

```
if ( Player->GetSpellEffects, "flame" == 1 )
    MessageBox "You have been flamed"
endif
```

This is the favorite possibility of adding new "spell effects". A dummy spell is created that does some minimal effect, e.g. raising luck by 1 point for 1 second. The GetSpellEffects function is used to detect if that spell has been cast on the player, and the script handles everything else. Sample script see below.

GetSpellEffects *does* work for race or birthsign abilities and powers (with of course the proviso that the power is currently affecting the calling object).

### RemoveSpellEffects, "Spell\_ID"

```
Player->RemoveSpellEffects, "flame"
```

Removes the effects of Spell\_ID from the player. This makes it even more useful for scripted spells.

```
if ( Player->GetSpellEffects, "flame" == 1 )
    player->RemoveSpellEffects, "flame"
    ;do what you want here
endif
```

## Managing and testing spell effects

### GetEffect, Effect\_ID (returns short)

```
If ( GetEffect, sEffectRestoreHealth == 1 )
```

This function returns TRUE if the calling Actor is being affected by the effect. Important: Effects are not spells, but the elements spells are made of. In the Appendix you can find a list of all spell effects.

Don't count on GetEffect being triggered immediately when a spell is added/cast: it will return 1 when the spell has begun to **affect** the actor, not when the spell is cast on or added to the actor (i.e. there's a short delay - might only be 1 frame but I didn't check).

**Note:** sEffectRestoreFatigue can't be detected using GetEffect. -(Phaedrus). This may also be true of other fatigue-related effects -(CaveRat).

sEffectWaterBreathing can't be detected from scripts, although it works in the console. Luckily there is a workaround in the form of the GetWaterBreathing function.

### RemoveEffects, Effect\_ID#\_enum

```
Player->RemoveEffects, 75
```

Removes all spells on the Actor that include the Effect. For this function you need the **number** of the effect-ID unlike the GetEffect function where you need the effect ID itself (Bravo, Bethesda!). Both can be found in the appendix.

**Important: Effects are not spells, but the elements spells are made of. In the Appendix you can find a list of all spell effects and their number.**

**Sample script:** This is a demonstration script that lets you check if a spell is in inventory, if it's active on the player, if the effect it causes is on the player and then removes the effect. Start it in the console using "StartScript Magictest" to try it out.

```
Begin Magictest

short var_1
short var_2
short var_3

if ( Player->GetSpell, "hearth heal" )
    set var_1 to 1
else
    set var_1 to 0
endif

if ( Player->GetSpelleffects, "hearth heal" )
    set var_2 to 1
else
    set var_2 to 0
endif

if ( Player->GetEffect, sEffectRestoreHealth )
    Player->RemoveEffects, 75 ;delete this line to see what happens normally
    set var_3 to 1
else
    set var_3 to 0
endif

MessageBox "GetSpell: %.0f    GetSpelleffects, %.0f    GetEffect: %.0f ", var_1, var_2, var_3

End
```

## Testing disease

**GetBlightDisease** (returns Boolean/short)  
**GetCommonDisease** (returns Boolean/short)

```
If ( Actor->GetBlightDisease == 1 )
```

Both functions return 1 if the calling Actor has the appropriate type of disease, otherwise 0. These are used in the disease scripts that give diseased or blighted creatures their disease:

## Sample Script:

```
Begin diseaseBlackHeart

DontSaveObject

if ( CellChanged == 0 )
    return
endif

if ( GetBlightDisease == 0 )
    AddSpell "black-heart blight"
endif

End
```

## Testing player blight disease (forum info / Rocket):

*Any time the current weather is blight, the player will have blight diseases added to them without any effects evident. They cannot be removed with cure blight spells and getBlightDisease will continue to return 1 for ever after unless they're removed by removeSpell. Next time you see blight weather though, they will be back. The engine picks random blight diseases, including any added by mods, with the exception of those that contain the corpus spell effect. You end up with them all very quickly.*

*As a guess, I would say it's a feature (according to lore, you can catch blight disease from exposure to blight storms) that is bugged and was subsequently disabled, without the code being removed. So we have this side effect. It actually breaks Bethesda's own game elements in that the Tribunal Shrines no longer give the message that you are not infected by blight when they should. The reason I suggest it is bugged is that the chance to catch blight disease from blight storms is 0.10 and yet you get them very quickly. Way too quickly. It would get annoying, even if the chance was reduced further. You catch it often enough just from the creatures as it is.*

*To disable this you need to modify the ini file. In the section [Weather Blight] you will find Disease Chance=.10 Set this to zero. If you then load a game where the player is currently in Red Mountain region, it appears that you will need to wait for a weather change to occur before the new setting takes hold (from blight to blight). Until then, you will continue to have them added by the engine.*

Further notes:

Auto-added blight diseases that aren't showing effects make the player effectively immune to those diseases, since the spell cannot be re-added without first removing it (use RemoveSpell).

The dialogue function PC Blight Disease is still a reliable test, as it only returns 1 if disease effects are present. I don't know of any reliable way of testing player blight disease by script without modifying the ini file, but GetSpellEffects can be used for individual diseases (there are only four besides Corprus in the unmodded game) if necessary.

## **Explosion**



### **ExplodeSpell "spellName"**

```
ExplodeSpell "proj_trap_spell"
```

The ExplodeSpell function makes a reference cast the given touch range spell at itself. If an area effect touch range spell is used, this can make the reference “explode”. See the TrapProjScript, shown in the tips and trick section on the Arrow trap.

## **Magic Get/Mod/Set effects functions:**

Most of these seem to refer to certain boni you normally get only from spells. With these functions you can apparently make them permanent or alter them. Most of these will normally use values between -100 to 100 (%) but will accept any number, but others are flags (0 or 1). Thus, you could make a creature that removes e.g. your ResistBlight bonus – wouldn't that be a nice surprise for our Nerevarine?

```
Get/Mod/SetResistMagicka
Get/Mod/SetResistFire
Get/Mod/SetResistFrost
Get/Mod/SetResistShock
Get/Mod/SetResistDisease
Get/Mod/SetResistBlight
Get/Mod/SetResistCorprus
Get/Mod/SetResistPoison
Get/Mod/SetResistParalysis)
Get/Mod/SetResistNormalWeapons
Get/Mod/SetWaterBreathing
```

Setting this to 1 enables water breathing

#### **Get/Mod/SetChameleon**

Corresponds to the Chameleon spell. This doesn't change the way the player looks (as a spell cast would), but has the same effect otherwise, i.e. NPCs may not detect the player and so on.

#### **Get/Mod/SetWaterWalking**

Setting this to 1 enables water walking

#### **Get/Mod/SetSwimSpeed**

#### **Get/Mod/SetSuperJump**

These correspond to the Swift Swim and Jump spell effects, so they normally range from 0 to 100, but work with negative or higher values as well.

#### **Get/Mod/SetFlying**

I found the following info on the UESP: This sets the player's flying mode. To get this cheat to work, enter the console command and then cast a Levitate Spell. The effect should now last until you disable the flying with the console (thanks Dave Humphrey).

#### **Get/Mod/SetArmorBonus**

corresponds to shield effect

#### **Get/Mod/SetCastPenalty**

corresponds to "Sound" effect? (<0 makes casting harder, >0, easier)

#### **Get/Mod/SetSilence**

#### **Get/Mod/SetBlindness**

#### **Get/Mod/SetParalysis**

What it does is each paralysis effect you apply to the actor increments the number by one. Each effect you remove decrements it by 1. You can also alter these using the set and the mod functions. Whenever its zero the actor can move. Try clicking on someone in the console and typing setparalysis 1. What's good about it is if you have multiple actors of the same ID, this allows you to paralyse an individual much like targeting them with a spell would. This is different from using a paralysis ability which would paralyse all of that ID if you left the cell then went in again. Paralysing them using SetParalysis lasts until you set it to zero or have gone out of their cell for 3 days even if they are not autocalc npc's. (Forum info / Cortex)

#### **Get/Mod/SetInvisible**

(Original MW: sic! Not invisible!)

#### **Get/Mod/SetInvisible**

(Later versions of MW, apparently spelling was fixed at some point (Forum info / Cortex))

#### **Get/Mod/SetAttackBonus**

corresponds to fortify attack effect

#### **Get/Mod/SetDefendBonus**

corresponds to sanctuary effect

## Sound

### *Make Actors speak an audio file*

**Say, "file name", "text"**

```
Actor->say, "vo\Misc\CharGenBoat1.wav", "This is where they want you."
```

Make subject "say" the sound file, only works on animating objects. The .mp3 voice sound files can be found in "Data files\Sound\Vo\" folder and are ordered in subfolders by race and gender. You can browse through most of them in the dialogue/voice window as well. Text is what is displayed as a subtitle as the file is played.

### **SayDone**

Returns true if the calling object is not saying anything.

Sample Script: from character generation:

```
begin CharGenBoatNPC
;this is the guard on boat who says to move along

short state
float timer

if ( menumode == 1)
    return
endif

if ( GetDisabled == 1 )
    return
endif

if ( OnActivate == 1 )
    return
endif

if ( GetDistance, Player < 180 )
    if ( SayDone == 1)
        ;first greeting
        if ( state == 0 )
            if ( timer == 0 )          ;using a timer so he doesn't talk ALL the time
                say, "vo\Misc\CharGenBoat1.wav", "This is where they want you.
Head down to the dock and he'll show you to the Census Office."
                set state to 10
            endif
            ;all other times after first
        else
            set timer to timer + GetSecondsPassed
            if ( timer > 6 )
                set timer to 0
                say, "vo\Misc\CharGenBoat2.wav", "Let's go. Move it along."
            endif
        endif
    endif
endif
end CharGenBoatNPC
```

### *Playing music*

**[no fix] StreamMusic, "filename.ext"**

Plays the sound file "filename.ext", usually an mp3 file, as the current music file. The music file should by default be located in the data files/music/ folder. Stream music can also play MIDI files (JOG).

**Note: Slightly bugged:** Calling StreamMusic automatically sets the music volume to 100, and leaves it there even after the music has finished playing. Since there is no function to set the volume, the user has to reset to his desired volume manually through the options menu.

## Playing sounds

[no fix] PlaySound, "sound ID"

[no fix] PlaySoundVP, "sound ID", volume\_enum, pitch\_enum

PlaySound3D, "sound ID"

PlaySound3DVP, "sound ID", volume\_enum, pitch\_enum

```
"ex_gg_portcullis_02"->Playsound3DVP "Dwemer Door Open" 1.0 1.0
```

The PlaySound function plays a sound without any modification. It does not matter to which object the script using the function is attached, the sound will always play at full volume, directly in the player's ear, so to speak.

The PlaySound3D function plays a directional sound source. The sound will seem to be emitted by the object to which the script with this function is attached.

The "VP" variants of each of these commands allow setting volume and pitch for the sound that is played. Bethesda has not made much use of this, it seems and it's only ever used with 1.0 set for both, which appears to be the standard anyway. My own experiments showed the sound not playing when I set volume to a variable, but this is no final verdict. The "sound ID" is the ID listed in the sound window accessed via Gameplay – sounds menu. You can add sounds there (place the .wav file somewhere in Data files/sounds). A good source for sounds on the web is <http://www.findsounds.com/>. Sounds should be in a certain format (see below) so you might have to change the format in a suitable program if you don't hear a sound in game.

## Controlling sound

StopSound, "Sound ID"

```
Object_ID->StopSound "Lava Layer"
```

stops the sound "SoundID" if it is currently playing in the calling object.

GetSoundPlaying, "sound ID" (returns Boolean/short)

```
if ( GetSoundPlaying "lava layer" == 0 )
```

Returns 1 when the specified sound is currently playing on the calling object. The sound ID's can be found in the Gameplay menu /sounds and /sound gen, where you can also set up your own (see below for formats). This function can be used to control sounds, but also to gain information, because certain sounds are tied to certain events in game, e.g. the "Critical Damage" sound or the "Disarm trap" sound.

**Sample Script:** This simple script ensures that lava always has its rumble sound playing:

```
begin lava
if ( menumode == 1 )
    return
```

```
endif  
if ( CellChanged == 0 )  
    if ( GetSoundPlaying "lava layer" == 0 )  
        PlayLoopSound3DVP "lava layer", 1.0, 1.0  
    endif  
endif  
end lava
```

## Sound file formats

**Note:** Not all sound files seem to play correctly in the game (although all play in the editor). To be safe, use the formats used by Bethesda (thanks to random name) :

"Cr" and the "Fx" folder  
Windows PCM (.wav)  
22050 kHz, 16-bit, Mono

Lower qualities work as well, e.g. 8,000 kHz; 8 Bit; Mono. Used e.g. in my "The Regulars" mod for Tavern music.

"Vo" folder format  
MPEG Layer-3, 64 Kbps  
44100 kHz, 16-bit, Mono

## Keeping track of time

There are a number of possibilities to follow the passage of time in scripts, including some that are not, or only poorly documented in the help file.

### **Timer**

**[no fix] GetSecondsPassed (returns float)**

A simple timer can be scripted with the GetSecondsPassed function. It returns the seconds passed *since the last frame* as a float value. To use this for a timer use something like the following example:

```
Begin TimerScript
Float timer
Short state

Set timer to (timer + GetSecondsPassed)

If (timer > 10)
    MessageBox "Displayed every 10 seconds"
    Set timer to 0
Endif

End TimerScript
```

**[no fix] GetCurrentTime (returns float)**

This returns the ingame time (24 hour) to two decimal places. This is exactly the same as using the global GameHour variable, but slightly less accurate.

### ***Morrowind's time related global variables***

**GameHour (is float global variable)**  
**Day (is short global variable)**  
**Month (is short global variable)**  
**Year (is short global variable)**

These globals get set by the game and contain the current date and time.

The MW calendar is a little bugged (thanks to samois for the info): MW starts on Day 16, Month 7, Year 427. (16 Last seed)

The months are as follows, with the days in each month.

(Morning Star ???)

Suns Dawn 31

First Seed 28

Rain's Hand 31

Second Seed 30

Mid Year 31

Sun's Height 30

Last Seed 31

Heart Fire 31

Frost Fall 30

Suns Dusk 31

Evening Star 30

So there are 334 days in a MW year!?! Basically it seems that Bethesda screwed up their code and lost a month, Morning Star / January... It seems the mistake was simply making it "wrap" to the wrong month from Evening Star. If you manually set month to 0 it will correctly display Morning Star in the rest menu. So this could be scripted around.

**Sample Script:** Checking the time of day with the GameHour function:

```
Begin AfternoonTea

If ( GameHour >= 17 )
    If ( GameHour <= 19 )
        "Cup of Tea"->Enable
    endif
elseif ( GameHour < 17 )
    if ( GameHour > 19 )
        "Cup of Tea"->Disable
    endif
endif

End AfternoonTea
```

## Keeping track of days passed

**Day (is short global variable)**

Use the global "Day" variable. It contains the current "day of the month" – so on "17, last seed" it is 17. This can be used to keep track of the number of days passed:

```
Short localdaysPassed
Short currentDay

if ( currentDay != Day ) ;whenever Day changes (presumably increasing)...
    set currentDay to Day
    set localdaysPassed to localdaysPassed + 1 ;add one to the counter
endif
```

This would usually be used for time-limited quests in a global script, to make sure the passage of time is correctly measured. Innovative scripting might also make use of it to trigger events after an item has been in the player's possession for some time, etc.



**DaysPassed (is short global variable)**

Contains the number of days since the game started. In order to work, DaysPassed has to be declared as a global short variable. This declaration is present in Tribunal.esm, but not in Bloodmoon.esm. Thus, in mods that make use of it, and that do not depend on Tribunal.esm, DaysPassed MUST be declared explicitly. This may be one of the reasons that some people reported it to be broken with Bloodmoon, but for others it worked fine - it depends on whether they had Tribunal.esm checked or not (Forum info / Erstam).

## Moon phases

Indispensable for any potential werewolf mod.

**[no fix] GetMasserPhase (returns short)**

**[no fix] GetSecundaPhase (returns short)**

```
If (GetMasserPhase == 4)
    [enable werewolf monster]
endif
```

**Note:** The helpfile lists *GetSecundusPhase*, the above syntax *GetSecundaPhase* is the correct one. Also note that *GetMasserPhase* and *GetSecundaPhase* return the value of the moon phases for the last exterior cell you visited (Forum Info / Elim).

I only made a quick test with these, but they seem to work.

Both functions return short with these values:

0 = MOON\_PHASE\_NEW (this is the default)

1 = MOON\_PHASE\_WAXING\_CRESCENT or MOON\_PHASE\_WANING\_CRESCENT:

2 = MOON\_PHASE\_WAXING\_HALF or MOON\_PHASE\_WANING\_HALF:

3 = MOON\_PHASE\_WAXING\_GIBBOUS or MOON\_PHASE\_WANING\_GIBBOUS:

4 = MOON\_PHASE\_FULL

# Weather

## Changing weather

[no fix] `ChangeWeather, "RegionID", short_Type_Enum`

```
ChangeWeather, "West Gash Region", 4
```

This function changes the weather in the indicated region to the weather type specified by `TypeEnum`, and will change again to according to the region settings after the time set by the game (I assume that is set in the `Morrowind.ini` file in the `Weather` section. In mine the entry reads:

Hours Between Weather Changes=20

The weather `TypeEnum` values are:

|   |          |
|---|----------|
| 0 | Clear    |
| 1 | Cloudy   |
| 2 | Foggy    |
| 3 | Overcast |
| 4 | Rain     |
| 5 | Thunder  |
| 6 | Ash      |
| 7 | Blight   |

## Changing weather settings for a region

[no fix] `ModRegion, "RegionID", clear_enum, [cloudy_enum], [foggy_enum], [overcast_enum], [rain_enum], [thunder_enum], [ash_enum], [blight_enum], [snow_enum], [blizzard_enum]`

```
ModRegion, "West Gash Region", 10, 20, 10, 5, 5, 40, 10, 0, 0, 0
```

Changes the weather chances for the `RegionID`. Used to get rid of, or add weathers to an area permanently. The values must add up to 100 or you will get odd results.

At least with `Bloodmoon`, all arguments except the first seem to be optional. I am not sure if this remains true with `Morrowind` and `Tribunal`.

If you want to maintain compatibility with a `Morrowind` only install, it would be best to use only 8 arguments.

## ***Determining current weather***

**[no fix] GetCurrentWeather (returns short)**

```
If ( GetCurrentWeather == 1 )
    ;[Do something if it is cloudy]
endif
```

This returns the weather TypeEnum listed above.

**Sample script:** Bethesda used this to make the banners move in the wind according to weather type:

```
begin OutsideBanner

;this script is for a banner object outside that
;animates in the wind.
;Idle is still, Idle2 is a little breeze, and Idle3 is a large breeze

short ran

if ( MenuMode == 0 )
    set ran to random 100
    if ( ran < 30 )                ;30% chance the flag does something new
        ;this will check the weather in the future
        if ( GetCurrentWeather >= 5 )                ;thunder, ash, or blight
            LoopGroup, Idle3, 5
        endif

        ;the last anim called in this script is the one it will play
        if ( ran <= 10 )
            PlayGroup, Idle

        elseif ( GetCurrentWeather < 5 )
            PlayGroup, Idle2

        endif

    endif

endif

endif
```

## ***Detecting wind speed***

**Undocumented:**

**[no fix] GetWindSpeed (returns float)**

I have only very briefly tested this function and it returns values, 0 indoors and floats outdoors (varying quickly, in overcast weather the values seemed to oscillate around 2). (Thanks to XPCagey for finding this)

GetWindSpeed will return a value greater than 0 in interiors acting like exteriors, which can be very useful if you need to detect an interior acting like an exterior.

**Sample Script:** This script detects if the players current cell is interior, exterior or interior acting like exterior.

```
if ( getInterior == 0 )
    ;outside
elseif ( getWindSpeed > 0.01 )
    ;Interior acting like exterior
else
    ;Interior
Endif
```

## ***Player sleeping***

### **[no fix] ShowRestMenu**

Brings up rest menu, and allows the player to sleep. This is used e.g. for beds in cells where it is otherwise illegal to sleep.

**Sample Script:** This is the standard script for beds:

```
begin Bed_Standard
;used for standard beds the player can activate and sleep in
if ( MenuMode == 0 )
    if ( OnActivate == 1 )
        ShowRestMenu
    endif
endif
end
```

### **[no fix] GetPCSleep**

**(returns Boolean/short)**

```
if ( GetPCSleep == 0 )
```

Returns true (1) if pc is sleeping. **Note:** The sleep selector and counter you see while sleeping counts as a menu. So be aware of that, if you want to use this function, and the MenuMode function in the same script!

The **example script** seems to come from a fairly useless item, but it demonstrates the use...

```
Begin pillowScript
short comfy
if ( PCVampire == 1 )
    return
endif
if ( comfy == -1 )
    if ( player->GetItemCount "misc_uni_pillow_unique" > 0 )
        if ( GetPCSleep == 0 )
            set comfy to 0
            return
        endif
    endif
endif
if ( comfy == 0 )
    if ( player->GetItemCount "misc_uni_pillow_unique" > 0 )
        if ( GetPCSleep == 1 )
            MessageBox "Your sleep is very restful on your Extra-Comfy Pillow"
            set comfy to -1
            return
        endif
    endif
endif
End pillowScript
```

### **[no fix] WakeUpPC**

Makes the PC wake up before the selected sleeping time is over. Sometimes creates a monster if the player was sleeping outside. This always happens if they try to sleep for only one hour, with longer times it may or may not happen (Thanks to Manauser for this info). WakeUpPC interrupts the rest only when you actually \*sleep\*. It does not affect loitering in places where resting is forbidden (Forum info / Kir).

**Sample script:** This is an edited excerpt from the lengthy "sleepers" script by Bethesda. It is responsible for giving you the dreams about Dagoth Ur that plague the player during the main quest. It shows how GetPCSleep and WakeUpPC can be used:

```
if ( GetPCSleep == 0 )
    return
endif

Set dream to 0

if ( GetPCCell "Balmora" == 1 )
    Set dream to 1
endif

if ( GetPCCell "Ald-ruhn" == 1 )
    Set dream to 2
endif
[...]
if ( dream == 0 )
    Set doOnce to 0
    ;this makes sure you have to leave the city and come back for another attack to occur
    return
endif

AddTopic "Disturbing Dreams"
;add this topic, doesn't matter if you do it over and over

;THE FIRST DREAM...

if ( GetJournalIndex A1_2_AntabolisInformant >= 10 )
    if ( GetJournalIndex A1_Dreams < 1 )
        WakeUpPC
        MessageBox "You had a disturbing dream. Bla bla bla", "Ok"
        Journal A1_Dreams 1
        return
    endif
endif
endif
```

## ***Enabling and disabling player control and interface***

### **Disable player control functions**

All of these functions disable some part of the user interface, thus restricting the players actions.

#### **[no fix] DisablePlayerControls**

Player can only look around with mouse or use options menu, nothing else and menus disappear.

#### **[no fix] DisablePlayerFighting**

#### **[no fix] DisablePlayerMagic**

These two functions seem to be unreliable according to forum information: If the player holds a weapon or has a spell readied he can continue to use it and quick-keys for weapons and spell likewise still seem to work. I currently don't know of a reliable solution for this problem.

#### **[no fix] DisablePlayerJumping**

#### **[no fix] DisablePlayerLooking**

#### **[no fix] DisablePlayerViewSwitch**

#### **[no fix] DisableVanityMode**

### **Enable player control functions**

Once a disable function has been used, the corresponding enable function can be used to restore control.

#### **[no fix] EnableLevelUpMenu**

[no fix] EnablePlayerControls (Enables the controls and menus.)  
[no fix] EnablePlayerJumping  
[no fix] EnablePlayerFighting  
[no fix] EnablePlayerLooking  
[no fix] EnablePlayerMagic  
[no fix] EnablePlayerViewSwitch  
[no fix] EnableRest  
[no fix] EnableVanityMode

### Check player control status

All of these functions return 1 if the corresponding "Disable" function has been called and is active, 0 if control is with the player.

[no fix] GetPlayerControlsDisabled  
[no fix] GetPlayerFightingDisabled  
[no fix] GetPlayerJumpingDisabled  
[no fix] GetPlayerMagicDisabled  
[no fix] GetPlayerLookingDisabled  
**GetPlayerViewSwitch**  
(**Broken**, function does not work. Instead use: )  
[no fix] GetVanityModeDisabled

### Force first or third person view

[no fix] PCGet3rdPerson (returns Boolean/short)  
returns 1 if in 3<sup>rd</sup> person mode

[no fix] PCForce3rdPerson  
queue the change to 3<sup>rd</sup> person mode (this may have to wait for the animation to finish)

[no fix] PCForce1stPerson  
same as above but 1<sup>st</sup> person mode

(See also the console command "ToggleVanityMode" (TVM).

### Functions for character generation menus

These undocumented functions are used for character creation. They enable all the menus that are used during the character creation process and enable certain basic features like the inventory, magic menu, stats window and the map:

#### Show character generation menus:

[no fix] EnableBirthMenu  
[no fix] EnableClassMenu  
[no fix] EnableRaceMenu  
[no fix] EnableNameMenu

There is no disable command for these. They are disabled by selecting ok in the menu.

#### Enabling in-game menus:

[no fix] EnableMagicMenu  
[no fix] EnableMapMenu  
[no fix] EnableInventoryMenu  
[no fix] EnableStatsMenu

Also no disable commands here unfortunately. These would have been useful.

The names of the functions should be self-explanatory. One use for these functions is as a cheat if you want to change your appearance or other things during a running game (although there might be problems associated with doing that). They can (and have been) used to create different ways of character generation. Be careful, sometimes these reset your level to 1.

**Sample Script:** This is one of many CharGen scripts that guide the player through character generation. This one is basically a safety feature for a player who just runs out the door, without triggering any or all of the little tutorials.

```
Begin CharGenDoorExit
;this is the door that exits the first part of the census building
;safety check for all menus on

short done

if (done == 1)
    return
endif

if ( OnActivate == 1 )
    enablestatmenu
    enableinventorymenu
    enablemagicmenu
    enablemapmenu
    enableplayerfighting
    enableplayermagic
    set done to 1
    Activate
endif

End
```

## ***Determining if player has menus open***

**[no fix] MenuMode**

```
If ( MenuMode == 1 )
```

MenuMode returns one if the player has menus open, and 0 otherwise. **This doesn't only apply to the inventory and dialogue menus.**

A good explanation is given by DinkumThinkum:

*It looks to me as though the game considers 'MenuMode' to be anytime the mouse pointer is on the screen instead of the crosshairs. I.e., any time you can move the mouse pointer around on the screen to select things, rather than the whole display moving and you can only select items at the center of the screen.*

*For example, hitting Escape for the in-game Options menu, displaying a message box with an "OK" button (or any other clickable buttons), hitting '~' for the console window: all those count as menu mode, in addition to the more obvious ones such as the dialogue window, the character data/map/inventory screen, the container window, etc.*

It is common practice to put the following lines at the beginning of almost any script, to prevent unnecessary or problematic functions being processed in MenuMode:

```
If ( MenuMode == 1 )
    Return
Endif
```

## ***Using MenuTest to open and close menus***

Undocumented:

**[no fix] MenuTest, short\_enum**

MenuTest doesn't return anything, however, when it is called, it closes certain types of inventory menus, including Player, NPC and containers. It doesn't work for dialogue, enchanting, alchemy, spell or armorer menus. (Forum Info / JOG, Jilin).

menutest or menutest 0 for closing menu

menutest 3 open stats menu or focus on it

menutest 4 open inventory menu or focus on it

menutest 5 open spell menu or focus on it

menutest 6 open map menu or focus on it

for menutest 3,4,5,6, it's like clicking on the upper right button of the menu

Example Script:

```
if ( OnPCEquip == 1 )  
    set OnPCEquip to 0  
    coc Balmora  
    MenuTest  
endif
```

## Miscellaneous functions and variables

### ***Breaking of script processing***

**[no fix] Return**

Return tells the game engine to finish processing the script for this frame. All code below this line will be ignored for this frame. In the next frame the script is executed again **from the top**.

```
If ( MenuMode == 1 )
    Return
Endif
```

Careful: Anything below a return function will not be processed, even if there are "true" if statements there! So use this with caution.

### ***Controlling global scripts***

These function are used to control global scripts. Global scripts have to be started with the StartScript function (either from another local or global script or from a dialogue result field) and a running script can be terminated with the StopScript function.

Tribunal "Start Scripts" in a plug-in start executing each time the game is loaded. If a Tribunal Start Script is terminated with a 'StopScript', it will start up again the next time the game is loaded - see the Tip and Tricks section on "Detecting when a player does a load from savegame". (Forum info / DinkumThinkum).

To my knowledge these functions do not work with local scripts attached to objects, you can however start and stop "targeted scripts" by using an object "fix": "ObjectID->StartScript" (for more info see the Tips and Tricks section on targeted scripts).

**StartScript, "ScriptName"**

This starts the specified script.

**Warning:** When you use startscript, all the scripts that have been run before you called startscript will be re-run, so scripts could be run more than once in the same frame.

**[no fix?] ScriptRunning, "ScriptName" (returns Boolean/short)**

The ScriptRunning function returns 1 if a script is running, 0 if it's not running:

```
if ( ScriptRunning, CharGen == 0 )
    StartScript CharGen
Endif
```

**StopScript, "ScriptName"**

Stops the specified script

**Note:** If you use 'StopScript' from inside the global script you're terminating, it doesn't actually terminate the script immediately. Instead, the script continues executing to the 'End' statement, and then terminates. So you still need to use "Return" if you don't want the rest of the script to be processed.

StopScript can also be used to construct do-once conditions for global scripts in a very clean way by self-terminating the script:

```
Begin do-once_script
[...]; do stuff here
StopScript do-once_script
End
```

### ***Fading the screen in and out***

**[no fix] FadeIn time\_float\_enum**  
**[no fix] FadeOut time\_float\_enum**

**[no fix] FadeTo alpha\_enum time\_float\_enum**

FadeTo 50 2.0 ;(Fades screen to 50% in 2 seconds)

FadeIn and Fadeout fades the screen (not an object) to blackness in the time specified (in seconds). Time is > 0 and <= 10.0. FadeTo fades only to a certain percentage: 0 is full transparency. 100 is black.

### ***Adding a location to the map***

**[no fix] ShowMap "cell ID"**

ShowMap "Gnisis"

This function will highlight the indicated cells. Cell ID can be full or partial, i.e. all cells that begin with the given string will be highlighted on the world map (e.g. ShowMap "Vivec" will highlight all the cantons).

**Sample Script:** reading this book will indicate all these places on the world map:

```
Begin bookPilgrimsPath
if ( GetJournalIndex TT_PilgrimsPath >= 100 )
    Return
endif
if ( OnActivate == 1 )
    Journal TT_PilgrimsPath 100
    ShowMap "Gnisis"
    ShowMap "Vivec"
    ShowMap "Ghostgate"
    ShowMap "Koal Cave Entrance"
    ShowMap "Fields of Kummu"
    Activate
endif
End
```

Also see "FillMap" console command.

### ***Assigning random values to variables***

**[no fix] Random, value\_enum**

Set my\_variable to Random, 50

Introducing some unpredictability into the effects of a script is a nice option, and it can be done with the *Random* function. *Random* returns values between 0 and the set value -1. So in the example above, my\_variable will be set to a value in the range from 0 to 49.

Note that the global short variable Random100 gets set each frame by the game's *Main* script to a random value between 0 and 100 (inclusive), so you can make use of that one, too.

Note that if you are using random100 in dialogue, you may want to add:

```
set random100 to random, 101
```

in the resultbox so that random100 is reset for the next topic chosen. Main does not set random100 in menumode.

**Note:** For any call to Random with a range over 100, the randomness of the return value gets very poor indeed... right up to Random, 255 where you only get 0 or 1... and any multiple of 256 also gets you a CTD. (Morrowind and Tribunal). In Bloodmoon, they seem to have fixed the randomness of the return value... you seem to get numbers that are more evenly distributed, even with a range above 100. But the CTD's at 256 and 512, etc, still happen (Info by Neko). It was furthermore discovered that sometimes the Random cap is set much higher than the number given. Setting any variable to a Random with the cap value of one of the following numbers produces some strange result, setting the higher cap actually to something around 1100: 65, 66, 68, 70, 71, 76, 77, 79, 82, 83, 84

TunaandCheese suggests that a method to work around the poor randomness in Morrowind and Tribunal is to merge Random results, as is demonstrated in this script:

```
begin randomnumber

;this script works out a random number between 0 and 10000

short spare
short number

;works out how many thousands there are
set number to random, 10
set number to ( number * 1000 )

;how many hundrerds
set spare to random, 10
set spare to ( spare * 100 )
set number to ( number + spare )

;and how many tens
set spare to random, 10
set spare to ( spare * 10 )
set number to ( number + spare )

;this has to be random, 11, rather than random, 10 as if it was random 10,
;it would only be 0 - 9999
set spare to random, 11
set number to ( number + spare )

;the variable number now contains a random number between 0 and 10,000 that you can use

end
```

## ***Playing videos***

[no fix] PlayBink "filename" flag\_enum

Pauses game and plays video. Set Flag to true if player can escape movie.

The video needs to be in Bink format and placed into the Datafiles/Videos directory.

To convert a video to Bink format you need to use [The RAD Video Tools](#).

## Levelled List functions



```
[no fix] AddToLevCreature    "levcreaname" "creature_ID" level_enum  
[no fix] AddToLevItem       "levitemname" "item_ID" level_enum  
[no fix] RemoveFromLevCreature "levcreaname" "creature_ID" level_enum  
[no fix] RemoveFromLevItem   "levitemname" "item_ID" level_enum
```

These functions are used to manipulate Leveled Item and Leveled Creature lists at run-time. Leveled lists are comprised of object/level pairs where the level is the level the PC has to be to encounter the object. The AddTo functions will add the given object/level pair to the specified leveled list as long as the list does not already contain a matching pair. The RemoveFrom functions will remove all occurrences of the object/pair from the leveled list.

**Additionally, if a RemoveFrom function is given an object pair with a level of -1, all object pairs containing the specified object are removed.**

**Note:** The RemoveFrom functions will not remove existing objects from the world. If a Leveled Creature reference has already calculated to be a certain creature, removing that creature from the Leveled Creature's list will not get rid of the existing creature in the world. However it will prevent that Leveled Creature reference from calculating to be that creature again.

**Note:** The ability of creature leveled lists to call other leveled lists is **only** in Bloodmoon. Those wishing to "nest" creature leveled lists in their plugins must have that Bloodmoon dependency, even if they use no objects from Bloodmoon. (forum info / blockhead)

**Warning:** It is not recommended that you use these functions as it will break any other mod that uses leveled lists.

DarkDragon writes on the Official forums:

*The problem lies in the way Morrowind loads items. First it loads ESMs, then ESPs, which can modify the leveled lists. If you have a list merger, you can merge those lists and keep the changes from all of them (where as before, the last loaded mod would take precedence). Then it loads ESS, or save files.*

*The problem is, if a script adds something to a Leveled List, it modifies the ESS (save game) and the save game then has a reference to that Leveled list. This is loaded last and it overwrites ANY changes done by other mods.*

### Sample Script:

When this script is placed on an object, activating it will toggle the existence of rats in the world by removing them from a Leveled Creature and removing rat meat from a Leveled Item.

```
Begin norats  
  
short norats  
  
if ( OnActivate == 1 )  
    if ( norats == 0 )  
        set norats to 1  
        RemoveFromLevCreature "rat_scamp_crab" "rat" -1  
        RemoveFromLevCreature "rat_scamp_crab" "rat-fast" -1  
        RemoveFromLevItem "lev_meat" "rat_meat" -1
```

```

        MessageBox "No more rats."
    Else
        set norats to 0
        AddToLevCreature "rat_scamp_crab" "rat" 1
        AddToLevCreature "rat_scamp_crab" "rat-fast" 1
        AddToLevItem "lev_meat" "rat_meat" 1
        MessageBox "The rats return."
    Endif
endif
end

```

## Square root



**[no fix] GetSquareRoot, number (float)**

```
set var_1 to GetSquareRoot var_2
```

The GetSquareRoot function returns the square root of the given number. This can be useful for vector or distance calculations (remember Pythagoras?).

It is possible to get the square root of a number without using this function, however it is significantly slower. See Soralis's Math mod for an example script. MWSE also has a square root function.

## Water Level Functions



**[no fix] GetWaterLevel (float)**  
**[no fix] SetWaterLevel newWaterLevel\_float**  
**[no fix] ModWaterLevel waterLevelChange\_float**

A great opportunity for cruel traps... These functions are used to determine and modify the Water Level of the current interior cell. When an Actor suddenly finds itself underwater, it will wait until it is halfway out of breath and then begin to make its way to the surface in a straight line up. Floating corpses are moved with the water without regard for collision.

**Note:** If you use GetWaterLevel in a interior cell with no water, GetWaterLevel will still return a result other than 0.

**Note:** Set/ModWaterLevel doesn't work in exteriors.

### Sample scripts:

This script goes on a crank to make it raise or lower the water level in the room.

```

Begin crank
short changelevel
float direction
float waterlift
short crankturn
short currcrank
float newwaterlevel

if ( MenuMode )
    return
endif

if ( OnActivate == 1 )
    if ( changelevel == 0 )

```

```

        if ( direction == 1 )
            set direction to -1
        else
            set direction to 1
        endif
        set changelevel to 1
    endif
endif
if ( changelevel == 0 )
    return
endif

set crankturn to 360 * GetSecondsPassed
set crankturn to crankturn * direction
set currcrank to GetAngle X
set crankturn to currcrank + crankturn
SetAngle X crankturn

set waterlift to 120 * GetSecondsPassed
set waterlift to waterlift * direction
ModWaterLevel waterlift

set newwaterlevel to GetWaterLevel
if ( direction == 1 )
    if ( newwaterlevel >= 600 )
        SetWaterLevel 600
        set changelevel to 0
    endif
else
    if ( newwaterlevel <= 0 )
        SetWaterLevel 0
        set changelevel to 0
    endif
endif
end crank

```

This modified “Float” script is placed on any object with a centered pivot point to make it float on the water’s surface regardless of water level. It also will make the object stop bobbing if the PC is standing on it.

```

Begin NewFloat

float timer
float swingTime
float startAngle
float currangle
short reset

float xvalue
float zvalue
float zoffset
float tmpoffset
float weightoffset

set startAngle to GetStartingAngle, x
if ( MenuMode == 0 )
    if ( timer == 0 )
        if ( reset == 0 )
            set timer to Random 100
            set timer to timer / 4
        endif
    endif

    set swingTime to 1

    set timer to ( timer + GetSecondsPassed )
    set currangle to GetAngle X
    set xvalue to 10 * GetSecondsPassed
    set zvalue to 5 * GetSecondsPassed

    if ( GetStandingPC )
        set zoffset to -30
    endif
endif

```

```

    SetAngle X 0
else
    ;rotate up
    if ( timer < swingTime )

        set currangle to currangle + xvalue
        SetAngle X currangle
        set zoffset to zoffset + zvalue

    ;rotate down
    elseif ( timer < (swingTime * 3) )

        set currangle to currangle - xvalue
        SetAngle X currangle
        set zoffset to zoffset - zvalue

    ;up again
    elseif (timer < (swingTime * 4) )

        set currangle to currangle + xvalue
        SetAngle X currangle
        set zoffset to zoffset + zvalue

    ;reset timer to zero
    else
        set timer to 0
        set reset to 1
        set zoffset to 0
        SetAngle, x, startangle

    endif
endif

set tmpoffset to GetWaterLevel
set tmpoffset to tmpoffset + zoffset

SetPos Z tmpoffset
endif
end NewFloat

```

## Tips and tricks

### ***Little helpers: Text search, copy, paste***

A good function to use for the beginning scripter is the search text function in the main TESCS edit menu. You can use it to search scripts, e.g. for a specific function you would like to use and want to find sample scripts for.

You can also use any text-editor, or the alternative scripting editors listed below for editing your scripts and copy / paste the script to and from the TES CS editor using ctrl-c / ctrl-v.

To make a copy of a script you want to change, don't just change the name, this will just overwrite the old script. Instead copy the original script (ctrl-a, ctrl-c) start a new script and paste the old one in (ctrl-v) now rename the script and change what you need.

### ***Alternative scripting editors***

1) MentalElf created an Elder Scrolls Scripting mode for the venerable **EMACS** coding editor, including automatic tabulation of if-blocks, and color coding of script code.:

<http://www.mentalelfz.com/>

EMACS is available for free under the gnu license (link is on MetalElf's page).

2) UESP's Dave Humphrey created **MWEdit**, an alternative construction set with improved scripting support (This is a beta version. I have only tested it briefly, but it looks very good, and seemed quite stable): <http://mwedit.sourceforge.net/>

Regarding scripting, Dave lists the following features:

- Color coding of script code. Use a default white or blue color scheme or use any custom colors to identify the various word types. Can also be disabled. (*see the trigonometry script below for an example of the color coding*)
- Select the font used in the script window.
- New script compiler reports many more errors and possible errors.
- Three levels of warning/error messages (weak, default, and strong) allowing you to set how many compiler messages are recorded.
- Compiler adds spaces to the script output where they might be required or expected (such as in if statements).
- Object types used in functions are checked more rigorously. If the function expects an NPC ID, you'll receive an error/warning if you use another type.
- Compiles script on save automatically (no warning/errors displayed).
- Export and import scripts to/from text files.
- View detailed help on all script functions.
- All compiler errors/warnings are displayed in a splitter pane at the bottom of the script window. Double-click a message to jump to the location of the message.
- View detailed information on compiler messages and display the function help if the message is related to a specific function.
- Compiler does not permit the use of reserved words as local variables (such as end, X, Y, etc...).
- Functions that are known to be broken will result in a compiler message.
- Simple function tooltips can be displayed for faster scripting.

## **Script Extenders**

Script extenders are programs that need to be run at the same time as Morrowind to work, they add new functions to the Morrowind scripting language.

This is by no means a guide to 3<sup>rd</sup> party programs; it is intended to give a brief overview. To find out more, it is recommended that you download the programs/dev kits.

A list of mods made using script extenders that should give an overview of what is possible:  
<http://www.mwmythicmods.com/MWE.htm>

Morrowind Enhanced (MWE):

<http://planetelderscrolls.gamespy.com/View.php?view=Utilities.Detail&id=8>

Adds many new functions, including getting references and using them with the vanilla Morrowind functions. It also adds better sound detection, skill leveling functions and functions to manipulate spells.

Scripts are written in the TESCS, with functions using unused commands like ToggleLoadFade. The esp then has to be processed by another compiler.

Unfortunately it is no longer being developed.

Morrowind Script Extender (MWSE):

<http://sourceforge.net/projects/mwse/>

MWSE is fairly similar to MWE in some respects, as it can get references, and then use those references with other default functions. It adds more functions than MWE, and is still being developed. It is slightly harder to code with, in that the MWSE functions cannot be used within normal if blocks, only within new ifx blocks, which have some drawbacks. However, the script readability is improved as it uses words for functions rather than cryptic unused commands and variable settings.

To compile MWSE scripts, you need to use MWEdit.

Morrowind Graphics Extender (MGE):

<http://www.tesnexus.com/downloads/file.php?id=5535>

MGE mainly allows manipulation of the rendering to the screen, so if you want to draw a texture to the screen, it is the program you want. It also allows you to apply shaders to the screen, press keys, manipulate the mouse, use bumpmaps and other graphics related features. It requires MWSE (Included in MGE), and scripts are compiled using MWEdit

## ***Script with style for safer scripting***

This is bound to be a bit controversial as it is as much about personal style as it is about fact. And it's bound to sound snobbish ☺. Nevertheless, I think this might be of some use for the newcomer, so here are a few comments about my personal views on good style and safer scripting:

- Use annotations. In a really short script they might seem useless, but even there you might want to state which mod/quest/item they belong to or what their general purpose is, etc. For long scripts this becomes indispensable, for yourself, if you stop working for a few days as well as for others that might want to learn from your script. Explain your variables, put headers on the main part of your script, comment on important lines of code, etc.
- Use state variables with style. Basically, due to the "executed once a frame" nature of the scripts this is your principal way of structuring your script for sequential events. There are several things to this.
  - A) Limit yourself to the minimal amount you need for the script.
  - B) use elseifs to chain different states of one state variable together, not separate if-blocks – this should be the main structural element of your script (it's not always possible nor necessary, but if it is, its going to save you a lot of trouble).
  - C) Check the elseifs are arranged from lowest to highest and in a logical order of events – will help you to keep everything organized, and thus avoids bugs. Jumping around wildly with state variables is the equivalent to careless use of GOTO in good old BASIC.
  - D) Use them extensively, do small steps. I can't tell you how many times bugged scripts started working simply because I moved some functions to a separate "state-block". Sometimes this doesn't seem to be logical at all – but if you can safely enter another step, do it.
- Decide for one style. TES Script is relatively forgiving regarding syntax. You can write functions small letters or capitalized as in this document, or all caps. You can use `if ( SomeFunction == 1 )` or `if (somefunction)`. But whatever you choose, try to be consistent in your usage.
- Use verbose variable names. A name that reflects the function of this variable makes a script much more readable. If you use global variables give them a unique name, e.g. put your initials in front or whatever – just try to minimize the chance that another mod will come up with the same name for a global – because that would screw things up royally.
- Keep track of your Return function uses. The Return function is inherently dangerous – remember that it will stop anything in that script below that line from being executed. Use it, but use it sparingly. If you have the feeling you have to use it a lot in one script, you should probably introduce a state variable instead.

## ***Cleaning up your mod***

When you work on your mod, you will probably want to look up other things as reference or simply to cut and copy things for your own purposes. The problem is that the TESCS will remember you looked at the things if you moved them the slightest bit in the game world, or if you ever hit an “OK” button, even if you did not change anything. Before you release your mod you should check it for such unwanted references and remove them.

In the File Menu select Data Files. Then select your mod like you normally would and hit the Details button. Look at the list of features here: this is a list of everything your mod changed or added. Look for things that you did not want to change, and select them.

Select a single item with Left Click.

You can select multiple items using Ctrl + Left Mouse Button.

If you click on an item and then use Shift + Left Click on another item, everything in between will be selected.

You don't have to select everything you want to remove at once, you can do it in multiple deletes.

To remove items, hit [del] on your keyboard. This marks the feature as ignored. Loading and resaving the mod will remove these ignored features from the mod.

An alternative, and much easier to use is the utility TESAME (TES advanced mod editor), available from various sites, e.g.:

<http://theseventhrealm.com/portal/tools.html>

A more recent tool that has become indispensable to me is the Morrowind Enchanted Editor, available at:

<http://tfo.rh.rit.edu/esforum/secretmasters/EnchantedSetup0.91c.exe>.

It's not very well documented, but offers a much nicer user interface than TESAME and is immensely powerful.

For scripts, try to remove any unused global variables or whole scripts you may have made in the course of developing your script. It bloats your .esp filesize, it probably wastes memory, or at the very least it looks bad.

### **This is a list of the change indicators found in the details tab of TESAME:**

DIAL - A new or changed topic

INFO - A dialogue response, or journal entry

REFR - a reference of an object that was put into the game world, while MISC, CONT etc. describe the actual Object (even if there is no instance of it placed in the world)

SOUN - A Sound

NPC\_ - A new type of NPC, or a changed NPC

CREA - A new creature or a change to a creature

LIGH - A new or changed Light

LTEX - An application of a landscape texture

PGRD - A change to the AI grid

CELL - obvious, signifies a changed cell either indoors or outdoors - in case your mod had nothing to do with that cell you should get rid of it. This affects all changes in that cell automatically, I think automatically

SCRPT - A Script - lots of people leave superfluous "test scripts" in their mods - that's bad style, I think.

MISC - A new or changed miscellaneous object - check names, delete if the change is unrelated to your plugin

ACTI - An activator - see above

CONT - A container - very critical: make sure you only change a new ID (copy of a container), not the original container - or they will all be changed.

STAT - A static object - see above

## On References Persist

The object windows in the TESCS have a checkbox named "References persist". Ticking this checkbox ensures that a reference (an instance of the object in the game world) is always available to be referenced by script, even if the player is in a different cell, or has not yet encountered the object.

If a script uses a specific reference to an object such as

```
RefObject->Enable
```

Then "RefObject" should have References persist checked.

Indirect referencing such as;

```
PlaceAtMe my_object 1 1 1
```

```
Player->AddItem my_object 1
```

do not usually require references persist. Some functions may be exceptional in this respect.

For example;

```
GetDistance, RefObject
```

requires that RefObject has been placed in the world and references persist checked.

Actors (NPC and Creatures) are always persistent.

(Thanks to Nigedo for additional information)

## The "72-Hours Bug": A Brief Explanation

*(Thanks to DinkumThinkum, Emma, Iudas, Qarl and others for this information.)*

The so-called "72-Hours Bug" is not really a bug at all - nor is it necessarily 72 game hours.

The time is set by fCorpseClearDelay (see "Game Settings"), which defaults to 72, and which also controls the delay before non-persistent corpses vanish. After the specified time, some temporary data is cleared to improve game performance.

Examples include:

- Talked to PC flag (dialogue) is reset.
- ForceGreeting: an actor will not be available for a ForceGreeting after the specified time unless "Corpses persist" is checked in the object info box.
- Actor stats and skills are reset to editor/autocalc levels.
- AI resets: e.g. an actor in combat with the player but with a low fight setting will no longer attack.
- Targeted scripts on non-persistent objects lose their targeting.
- OnActivate may be reset (see function description for more details).

Specific workarounds are mentioned in the relevant function's description. Alternatively, other methods can be used to store information. For example:

- A script that runs at least once each load session can be used to keep track of information, e.g. a companion's stats and skills.
- Journal entries or global variables can be used to store information, e.g. as a substitute for the Talked to PC flag.
- Token items may be added to an actor's inventory (or a refs persist container).

Of course, other workarounds may be devised. Note that it is usually not advisable to change the value of fCorpseClearDelay, as GMSTs are not accessible to script and your change may break mods which rely on time-dependent methods of avoiding the loss of temporary data. (Changing the setting to a *higher* value is less likely to break other mods, but more likely to cause CTDs for those whose system can't cope with the amount of data stored.)

## Limits of the Script Editor

**Character Limit:** There is a limit of the maximum number of characters per script. It is somewhere around 30000 characters ( the true limit is most likely 32767, which is the max value for a 16-bit signed integer, which is how script length is stored in the .esp – thanks to Horatio for this info). If this occurs you can no longer type in the editor window. To save characters try the following:

- Remove characters
- Use shorter variable names
- See if the script can be split and some part maybe handled in a global script or attached to a separate object as a separate script.

**Line Limit:** there also is reportedly a maximum line limit. This seems to vary and reports on the forum range between 900-and 1500 lines of code. It's probably rather a limit of the compiled script than the actual lines of text, so empty lines and comments don't count. This is reported by an error message upon saving the script.

**If-elseif limit:** There is a limit on the maximum number of if-elseif conditions that can be used per script. I am not sure of the absolute number (I heard both 127 and 256). Also there is a maximum depth of nested if commands, it's reportedly 10 (thanks Riiak). *(Note: Error message for exceeding the maximum depth of nested if's is "Max nesting of 10 exceeded on line XXX" - in this case, you'll need to break it up somehow.)*

**Instruction Limit:** An If-block or While-block is limited to 255 compiled commands (per block, not per script). The limit is *not* on the number of lines (each line may end up as several commands when compiled), and blank lines and comments don't count towards this total. - (Dave Humphrey, thanks to Galsiah for pointing this out)

**Script Name Limit:** TunaandCheese reported a limit to the length of the scriptname: Your script's name should never go over 31 characters in length. If you have 32 characters in the script name there will be a box on the end of the script and when there is 33 characters the 33rd character goes funny, the 34th character seemed to be fine. Character 32 gets replaced by some other character in the title bar and in the Details list information. From looking at the .esm/.esp/.ess format information on Argent's web site, apparently only 32 bytes are allocated for the script name, which makes a 31 character limit reasonable. -(DinkumThinkum)

## ***Pitfalls***

### **Inconsistent Commas**

The scripting language is fairly forgiving in terms of comma usage or not, but the inconsistent use of commas can cause difficulties:

This will work:

```
Player->PositionCell, -1396, 124, 3312, 90, "Cell ID"
```

and this will work:

```
Player->PositionCell -1396 124 3312 90 "Cell ID"
```

but this will sometimes have problems:

```
Player->PositionCell -1396, 124, 3312, 90 "Cell ID"
```

### **Object Names**

Don't start object names with an underscore. Otherwise, some script operations will fail.

This will cause runtime error.

```
_wr_testNPC->forceGreeting
```

This will work fine

```
wr_testNPC->forceGreeting ;--Okay.
```

### **Fractional Numbers**

A constant float specified without digit before the decimal point will cause a runtime error.

Floats in the range:  $-1 < 0 < 1$  should be specified with a leading '0'. Otherwise you'll experience a runtime error.

This gives you a runtime error

```
if ( number == .2 )
```

This runs fine

```
if ( number == 0.2 )
```

### **34th Variable**

If you have a script with 34 or more variables of the same type (short, long or float), the 34th variable of that type can cause errors when used. For that reason, most scripters call it "DoNotUse" or something like that to remember that they should not use it. The reason for this problem seems to be that the ASCII character 34 is the double quote character.

```
Short 32ndVar
Short 33ndVar
Short DoNotUse ;Really the 34th Var
Short 34thVar
Short 35thVar
```

## ***Saving CPU time***

If you plan a mod with lots of scripts or long and involved scripts, you may want to give some thought to not wasting CPU power. There are a number of things you can do here:

If the script does not have to be executed every frame, **put a little counter in there:**

```
Begin My_super_long_script

Short framecounter
If ( framecounter < 10 )
    set framecounter to ( framecounter + 1 )
    Return
Endif
set framecounter to 0

[super long script goes here]
end
```

This little piece of code, that should be at the very top of your script, will allow your script (or rather the main, CPU power eating part of it) to be executed only every 10<sup>th</sup> frame. You could do the same thing with a timer, and execute the script only every 3 seconds or once every minute.

**Execute script only if the player is suitably near.** If you have scripted a fancy magic bouncing ball, or basically anything that is a visible effect, there is no reason to run the script if the player cannot see it. So put something like this on top of your script:

```
If ( GetDistance, player < 5000 )
    Return
Endif
```

However, as `getDistance` isn't an amazingly fast function, you wouldn't want to put this check above a simple `getScale`, `setScale` script.

**Shortcut scripts that are no longer needed.** If you have local scripts that you may not need from a certain point onwards, e.g. because of Actor death or because the object was disabled, reduce their CPU need by putting something like the following at the top of the script:

```
if ( GetDisabled == 1 )
    Return
Endif

If ( GetHealth <= 0 )
    Return
Endif
```

**Terminate global scripts.** Remember, global scripts are running all the time until you stop them again with `StopScript`. You can do do-once global scripts by just putting a `StopScript` command at their end

```
Begin do_once_global_script

[your code here]

StopScript "do_once_global_script"

End
```

**Try to use local scripts** instead of global scripts. With local scripts you are sure that they only run when you are in the vicinity. Think hard before making a script global if it can be done with a local script instead.

**Be careful with while-loops, GetDetected, GetLOS** and other "slow" functions. Use methods as described above (e.g. a counter or a timer) to make sure they are not called too often.

**Stop script while in menu mode.** Always, unless you have specific reason not to, put the following at the top, to avoid the mouse lagging in the menu and other unwanted effects.

```
If ( MenuMode == 1 )  
    Return  
Endif
```

**In general, reducing the number of scripts running improves efficiency** more than reducing the complexity of individual scripts.

So, for example, if you have two short globals running all the time, it is better to merge them into one global.

If you've got a few global scripts which only need to run based on some check, it makes sense to place the checks in one "parent" script, and startscript the others from there when they need to run (remembering to ensure to run them once per game session if they need to conserve variable values).

## **Targeted scripts: running "global" scripts tied to an object**

`"Object_ID"->StartScript "Script_ID"`

*(Credit for discovery of this technique and much of the following information goes to FreshFish. Further information supplied by Riiak, MentalElf, DinkumThinkum, Argent, Cortex, and others.)*

It is possible to use the StartScript function to run global scripts that are tied to an object or Actor. These scripts resemble both local scripts (in that the functions called always default to the object or Actor the script targets) and global scripts (in that they are always running). Objects may have several different targeted scripts running on them at one time, and may also have a local script.

### **To start a script as a targeted script:**

- `Object_ID->StartScript Script_ID` (may be used from script, but does not work from dialogue results).
- Start the script from the intended target's local script or another targeted script on the object (it will inherit the target).
- Start the script from dialogue results (this will only work to target the actor the player is in dialogue with: it is not possible to specify a different target in dialogue results). This method is particularly useful as you don't need access to scripts running on the actor, nor do you need an ID.

### **Uses of targeted scripts:**

As a general rule: If a function requires a fix in a global script but the fix can be omitted in a local script, that function may be used without a fix in a targeted script as well. Examples of popular functions for targeted scripts include:

`AddItem, RemoveItem, GetItemCount`  
`AIFollow, AIWander, AIEscort`  
`GetPos, SetPos, PositionCell`  
`Get/Set/ModStat`  
`StartCombat, StopCombat`

Targeted scripts can be powerful tools, especially in combination with dialogue: you can create generic dialogue (e.g. filtered for class only) and start your script from the resultbox. CDCooley's "Companion Teleportation" is a good example of use with dialogue. Voice dialogue can also be used this way: for example, many companions have a targeted "pacifist" script that is started from Attack voice results if the player tells the companion to avoid combat.

### **Cautions and limitations:**

- Variables defined in a targeted script are not considered local to the object from the point of view of dialogue and other scripts. For example, they cannot be used as dialogue conditions. The special variable "companion" also cannot be used, i.e. declaring a short variable "companion" in a targeted script and setting it to 1 will not enable companion share on the target actor. I assume this applies to other special locals as well, but I haven't tested any others.
- You cannot have more than one instance of a targeted script running at one time (same as global scripts). The obvious workaround is to have several different scripts that all do the same thing.

- If OnActivate is used in a targeted script, the object will not be able to be activated by normal means after the script is stopped, unless a targeted script with an OnActivate is again added to the object.
- If an object is deleted while a targeted script is running on it, the game will CTD. If the target is not necessary but is merely a consequence of the way the script was started, you can usually target the script on the player instead to avoid this: "player->StartScript ScriptName". Otherwise, the script must be stopped before deleting the object.
- Scripts will detach from non-persistent targets when the game is saved and reloaded. This may give errors on load ("Unable to locate reference for global script..."), or may not; either way it may also give odd effects in some circumstances.
- When a targeted script is started on an object that was not placed into the gameworld in the editor (i.e. an object that was placed or generated during the game), the script will lose its targeting when a savegame is reloaded. Note that this applies to the player character as well.
- Load list changes may also cause targeted scripts to lose their targeting when the game is reloaded, whether or not the target is persistent and whether or not it was placed in the editor. This doesn't always happen, but it can (adding several mods higher in the load list will usually do it).

If a script becomes detached from its target, some functions that manipulate data about the previously targeted object (e.g. getHealthGetRatio) may cause CTD if used without a fix.

As a workaround to avoid untargeted scripts running, you may be able to use a startscript to stop the targeted script if it is running, then restart it if necessary. As a safety check for targeted scripts on actors, GetHealth may be used before any functions that may cause problems (returns 0 in an untargeted script):

```
if ( GetHealth <= 0 )
    StopScript ScriptName
    return
endif
; main body of script goes here
```

### ***Detecting when the player does a load from saved game:***

Some reasons to test for player loading a saved game:

- 1) To continue custom music (mp3 type music resets on load).
- 2) To keep a NPC running or sneaking.
- 3) To reset an object to its proper scale (if outside of range 0.5 to 2.0).

There are a number of ways to do this:

JOG proposed the use of SetJournalIndex:

```
if ( ( getjournalindex "dummy" ) != 100 )
    MessageBox "You just reloaded, Cheater!!!"
    setjournalindex "dummy" 100
endif
```

"Dummy" is any journal-topic that has no text for index 100.

SetJournalIndex will set the index to the new value, no matter if an entry exists for this value or not, but when you reload, the index will be reset (see SetJournalIndex for more information).

MentalElf suggested that GetForceRun, GetForceSneak, GetScale can all be used to detect when the player has just loaded a save game. This is due to ForceRun and ForceSneak being cleared during a load from saved game, and scale is set to within the range 0.5 to 2.0. In my opinion ForceRun is best, as ForceSneak puts the NPC into a crouch posture.

```
; (NPC object)
if ( GetForceRun == 0 )
; Player just loaded a saved game
; Handle game load here.
ForceRun
Endif
```

A different option is to use start scripts ( available only with Tribunal and Bloodmoon): Here are two examples by Dinkum Thinkum:

```
begin DT_DoOnce_TribStartScript02

;script to demonstrate a 'DoOnce' Tribunal Start Script that runs once
; each time a saved game is loaded with the mod enabled
;by DinkumThinkum

;script executes 'Do Once' code section once
; each time the mod is loaded as part of a game

;start of 'Do Once' code to be run each time a saved game is loaded with this mod enabled.
MessageBox, "You will see this message everytime you load a saved game with this mod
enabled.", "OK"

;end of 'Do Once' code.

StopScript DT_DoOnce_TribStartScript02
end DT_DoOnce_TribStartScript02
```

```
begin DT_DoOnce_TribStartScript01

;script to demonstrate a 'DoOnce' Tribunal Start Script
; by DinkumThinkum

;script executes 'Do Once' code section once, when the mod is loaded
; with a saved game that does not include the mod (or with a new game)

;script executes 'Reload' code section once, each time a save game
; made with the mod is reloaded with the mod enabled

;either code section can be omitted, but all the control structures will still be needed

;Note: BOTH 'StopScripts' are necessary. Trust me... 😊

;DT_DoOnce_TSS01 - Global variable, won't be reset by StopScript (initialized to 0)
if ( DT_DoOnce_TSS01 == 1 )

;start of 'Reload' code, to be run once each time mod is reloaded
MessageBox, "You have loaded this mod with a saved game made with the mod enabled.", "OK"

;end of 'Reload' code

StopScript DT_DoOnce_TribStartScript01
Return
endif
```

```
;start of 'Do Once' code, to be run only when the mod is first loaded.

MessageBox, "You have loaded this mod with a saved game that was made without this mod
enabled.", "OK"

;end of 'Do Once' code.

set DT_DoOnce_TSS01 to 1
StopScript DT_DoOnce_TribStartScript01

end DT_DoOnce_TribStartScript01
```

### ***Uses of the CharGenState variable - Disabling saving and menus***

The availability of the save option in the main menu (and the Quicksave key) depends on the value of the CharGenState global. Set it to something other than -1 (to 99, for example), and the player is no longer allowed to save the game. Setting CharGenState back to -1 again turns everything back to normal.

This has several side effects. First, you won't be able to access the menu screen. This can be fixed by EnableStatsMenu - it reenables ALL the menus, not only the stats window. However, the journal, the Quick Keys and the QuickKey menu (F1) are disabled as well, and you can't loot corpses. I haven't found a way around these limitations.

There is some conflict potential in this, so it would be good to take some precautions that the player is not currently inside the character generation sequence, and that the value is set back to its original value after use.

(Forum info / Erstam)

#### **Note:**

If you use a command like EnableStatsMenu or EnableInventoryMenu while CharGenState is other than -1, you will never be able to turn off the menus again with CharGenState. The EnableInventoryMenu command only seems to have this effect in the CharGen process, while EnableStatsMenu has this effect all the time. If the menus were supposed to be turned off in a mod, it won't work. You still can't save or use quick keys. But what really is annoying about this is that this applies for the whole game session! You actually need to exit the game and restart it, so you can never rely on that menus will be turned off.

(Forum info / Björn )

## ***Detecting use of scrolls or books***

This is a surprisingly difficult task, as OnActivate and OnPCEquip are both needed AND don't work quite as expected. Kir has found a solution as shown in this script for invoking a letter of credit:

```
Begin BankLetter10

short button
short messageOn
short invoke
short gone
short goneway
short testdist
short PCSkipEquip
short OnPCEquip

set PCSkipEquip to 1
; disable if used (a 1 use item)
if ( gone == 1 )
    if ( goneway == 1 ) ; activated as external item
        Disable
    else ; equipped from inventory
        startscript BankLetter10Remove
    endif
    set gone to 0
    return
endif

if ( OnActivate == 1 )
    Set messageOn to 2
    set goneway to 1
endif

If ( OnPCEquip == 1)
    Set messageOn to 2
    Set OnPCEquip to 0
    set goneway to 2
endif

if ( messageOn == 0 )
    return
endif

if ( messageOn == 2 )
    MessageBox "Do you want to invoke the Letter of Credit?" "Yes" "No"
    Set messageOn to 1
    return
endif

if ( messageOn == 1 )
    set button to GetButtonPressed
    if ( button == 0 )
        Set invoke to 1
        Set messageOn to 0
        return
    ;
    endif
    if ( button == 1 )
        Activate
        Set messageOn to 0
        return
    endif
endif

if ( invoke == 1 )
    PlaySound "Item Gold Up"
    Player->AddItem, Gold_001, 10000
    set gone to 1
    set invoke to 0
endif

End
```

Erstam posted a script with even better features, that also revealed an interesting glitch with the scripting variables OnPCEquip / SkipEquip:

"Inspired by the BankLetter script in MSFD 7, I have found a way to run custom script code on books and scrolls when either "equipped" from the player's inventory or activated from the game world, *while the book/scroll is displayed as normally*. Surprisingly, it's the PCSkipEquip variable that is set to 1 when the book is dropped on the player's portrait, rather than the OnPCEquip variable. This is the code I used:"

```
Begin activateBook

short OnPCEquip
short PCSkipEquip
short doOnce
short actionFlag

if ( actionFlag == 1 )
    if ( doOnce == 0 )
        ; insert your custom code here
        set doOnce to 1
    endif
    set actionFlag to 0
endif

; PCSkipEquip is set to 1 every time the book is equipped from your inventory
if ( PCSkipEquip == 1 )
    set PCSkipEquip to 0
    set actionFlag to 1
    return
endif

; these lines are important, otherwise the book can't be picked up from the ground
if ( MenuMode == 1 )
    return
endif

; for activating the book when it is placed in the game world
if ( OnActivate == 1 )
    set actionFlag to 1
    Activate
endif

End
```

It should work without the doOnce condition, in case you want the action to take place every time you equip the book, but I haven't tested it yet.

## ***Making Actors switch between weapons***

Since the equip function does not work, the only way to do this is to take items away from the Actor, or to modify his skills at runtime.

Here is an example I used to make a guard switch between bow and sword:

```
Begin HBCaravanGuardAI

; this script makes the AI guard more dangerous by making him switch from bow to sword when
the player closes in
short currentarrows
short storearrows
short doonce

set currentarrows to GetItemCount "arrow of wasting flame"

if ( doonce == 0 )
    set storearrows to currentarrows
endif

if ( GetDistance, Player < 120 )
    set currentarrows to GetItemCount "arrow of wasting flame"
    if ( currentarrows > 0 )
        RemoveItem "arrow of wasting flame", 1
        set doonce to 1
    endif
elseif ( GetDistance, Player >= 120 )
    if ( currentarrows < storearrows )
        AddItem "arrow of wasting flame", 1
    else
        set doonce to 0
    endif
endif

End
```

The next example is one by Bethesda, which does the same thing, using the skill change method (admittedly more elegant than mine 😊):

```
begin marksmanToggle

short counter
short myMarksman

if ( MenuMode == 1 )
    return
endif

if ( counter < 20 )
    Set counter to counter + 1
    Return
endif
if ( myMarksman == 0 )
    set myMarksman to GetMarksman
endif
if ( GetMarksman > 0 )
    if ( GetDistance Player < 400 )
        SetMarksman 0
    endif
else
    if ( GetDistance Player > 600 )
        SetMarksman myMarksman
    endif
endif

;for level designers... forces AI to do what it ought to do
;when they are near, they use melee weapons
;when they are far, they use missile weapons
;checks every 20 frames for speed
;Note: does not affect spellcasting AI

End
```

## ***Making NPCs switch between spell "sets"***

If you want an NPC to switch between different sets of spells, the skill-setting method will not work unless the spells are grouped into different schools (which is unlikely to be the case). Adding and removing spells will only take effect when the NPC is not in combat, but StopCombat will stop combat for all actors involved (usually not a good idea). However, other AI commands can be used to stop combat for a single character. This method is more useful for followers of the player than for hostile NPCs as the character will exit combat, however briefly: since the enemy will generally attack the player rather than a follower, temporarily removing a follower from combat is unlikely to cause problems; using AIFollow to remove the follower from combat ensures that the NPC will immediately re-enter combat at the first hit by or on the player. AIWander (and probably other AI commands) will also work, but you may need to restart combat using StartCombat afterwards.

The following example is an abbreviated snippet from the local script of a companion I'm working on. Most of the time the spell-switching is barely noticeable (just a moment of hesitation), and the companion quickly re-enters combat.

```
;CombatStyle is set from dialogue when the player gives combat orders
if ( CombatStyle == 1 ) ;switch between targeted and touch spells
    if ( player->GetSpellReadied == 1 )
        set CombatMage to 1
    elseif ( player->GetWeaponType >= 9 )
        set CombatMage to 1
    elseif ( player->GetWeaponType <= 8 )
        set CombatMage to 2
    endif
elseif ( CombatStyle == 2 ) ;switch between targeted spells and weapons
    if ( player->GetSpellReadied == 1 )
        set CombatMage to 1
    elseif ( player->GetWeaponType >= 9 )
        set CombatMage to 1
    elseif ( player->GetWeaponType <= 8 )
        set CombatMage to 3
    endif
elseif ( CombatStyle >= 3 ) ;no combat magic
    set CombatMage to 3
elseif ( CombatStyle == 0 ) ;AI choose
    set CombatMage to 0
endif

if ( CombatMage != MageState ) ;MageState is a doonce for spell switching
    if ( MageState != 3 ) ;don't remove spells that aren't there
        if ( CombatMage != 0 ) ;don't remove spells just to add them back
            ;remove unwanted spells here
        endif
    endif
    if ( CombatMage == 0 )
        ;add all combat spells here
    elseif ( CombatMage == 1 )
        ;add targeted spells here
    elseif ( CombatMage == 2 )
        ;add touch spells here
    endif
    set MageState to CombatMage
    AIFollow player 0 0 0 0 ;this ensures that changes take effect immediately
endif
```

## ***Making Actors lie down***

It's not uncommon for modders to want actors to lie down, e.g. as a scheduled rest at night or a "corpse" that comes to life to attack the player. Making this happen is not as simple as it sounds, so I thought I'd present a few ideas here. **This is not a complete list of all information:** it's just a place to start.

**Custom animations:**

These can be added to NPCs in the Construction Set (open the NPC's info box, "Add animation file" button, and select file); to play the animation use AIWander with 100 chance for the relevant idle and 0 chance for all other idles (assuming the animation is set to replace an idle, but most are).

Note that custom animations replace at least one standard animation, so you will also need to ensure that animation isn't played at random when you don't want the actor lying down. You will also need to prevent the player activating the NPC while the animation is playing (since the NPC is alive and conscious, activation will give normal dialogue), and prevent the NPC saying voice entries: this can be achieved by adding an empty entry for each relevant voice dialogue type and filtering it carefully. You can also add an empty greeting (carefully filtered) to prevent the player talking to NPCs, or creatures with dialogue.

**PlayGroup:**

In theory at least, you can use knockdown or death animations to make an actor lie down (for basic usage, see the function description). Note that some character animation groups may not work as expected with Bloodmoon, and some are different for different races. Note also that the NPC's upper body may not play the correct animation; this may be version-dependent (not tested) so even if it seems to work - be careful! In my experience, under version 1.6.1820, PlayGroup is too unreliable to be of much use with NPCs.

If you do use PlayGroup, you will also need to stop NPCs using normal voices and prevent the player activating them for dialogue, as above.

**0 Fatigue:**

This is usually the easiest method to use, and you won't need to do anything special to prevent the actor talking to the player while unconscious/dead/asleep. Note however that it will not work reliably by itself: usually the actor will not fall unless it takes a hit in combat or tries to move from where it is standing. Also note that if you use ModCurrentFatigue, the actor's Fatigue may end up set to a negative value: if you want to revive the actor at a particular time, be sure to mod Fatigue back up completely.

From my own tests, AIActivate is the best and most reliable method of getting the actor to move (and therefore fall) when at 0 Fatigue. Place the object to be activated far enough away from the actor so that the actor will try to move, and give a different AI command when reviving the actor. AIWander with distance (e.g. "AIWander 512 0 0") may also be useful, but be aware that the actor may stand in place for a while before falling (depends on the random idle/wander).

Other AI commands (including AIFollow) are not suitable: for example, AIFollow will have the actor standing up to follow when necessary, then falling down when the follow target stops moving. AIEscort *may* work if you can be sure the escort target will never be too far away: once the target is far enough away that the actor would normally stop and wait, the actor will "warp" to a standing position and stay there until the escort target comes back within range.

If you want the NPC to stop breathing, you can wait briefly after the AIActivate to give your NPC time to collapse, then add a paralysis ability.

## Arrow- or magical traps

I had originally posted this as an example script for setdelete (as which I received it from Bethesda), but I think it is better placed here.

When this script is placed on an object, as soon as it is placed in the world (or encountered if it was placed in the editor) it will calculate where the player is and move towards that location at a steady rate (determined with GetSquareRoot). When that point is reached, or if the object is ever within range of the player, it explodes (with ExplodeSpell) and disables. Once it has detonated, it waits a few frames for the spell system to clear, then deletes itself (with SetDelete).

```
Begin trapProjScript

short range
short initialized
short distance
short detonate
short triggered

float targx
float targy
float targz
float shiftx
float shifty
float shiftz
float currshift
float currx
float curry
float currz
float totaldist
float rate

float killtimer

if ( triggered == 1 )
    if ( killtimer < 4 )
        set killtimer to ( killtimer + GetSecondsPassed )
    else
        SetDelete 1
    endif
    return
endif

if ( MenuMode == 1 )
    return
endif

if ( initialized == 0 )
    set initialized to 1
    set range to 150
    set rate to 300
    set targx to ( player->GetPos X )
    set targy to ( player->GetPos Y )
    set targz to ( player->GetPos Z )
    set shiftx to ( targx - GetPos X )
    set shifty to ( targy - GetPos Y )
    set shiftz to ( targz - GetPos Z )
    set totaldist to ( ( shiftx * shiftx ) + ( shifty * shifty ) + ( shiftz * shiftz ) )
    set totaldist to GetSquareRoot totaldist
    if ( totaldist != 0 )
        set shiftx to ( shiftx / totaldist )
        set shiftx to ( shiftx * rate )
        set shifty to ( shifty / totaldist )
        set shifty to ( shifty * rate )
        set shiftz to ( shiftz / totaldist )
        set shiftz to ( shiftz * rate )
    else
        set triggered to 1
        return
    endif
endif

endif
```

```

set distance to GetDistance "player"
if ( distance < range )
    set detonate to 1
else
    set currx to GetPos X
    set curry to GetPos Y
    set currz to GetPos Z

    set currshift to ( shiftx * GetSecondsPassed )
    set currx to ( currx + currshift )

    set currshift to ( shifty * GetSecondsPassed )
    set curry to ( curry + currshift )

    set currshift to ( shiftz * GetSecondsPassed )
    set currz to ( currz + currshift )

    if ( shiftx < 0 )
        if ( currx < targx )
            set detonate to 1
            set currx to targx
        else
            set detonate to 0
        endif
    else
        if ( currx > targx )
            set detonate to 1
            set currx to targx
        else
            set detonate to 0
        endif
    endif

    if ( shifty < 0 )
        if ( curry < targy )
            set detonate to 1
            set curry to targy
        else
            set detonate to 0
        endif
    else
        if ( curry > targy )
            set detonate to 1
            set curry to targy
        else
            set detonate to 0
        endif
    endif

    if ( shiftz < 0 )
        if ( currz < targz )
            set detonate to 1
            set currz to targz
        else
            set detonate to 0
        endif
    else
        if ( currz > targz )
            set detonate to 1
            set currz to targz
        else
            set detonate to 0
        endif
    endif

    SetPos X currx
    SetPos Y curry
    SetPos Z currz
endif

if ( detonate == 1 )
    ExplodeSpell "proj_trap_spell"
    set triggered to 1
    disable
endif
end

```

## Scripted teleporting

Teleporting to variable positions in interior, and especially exterior locations is not trivial, there are issues with surrounding cells not loading properly (meaning part of the landscape may not be rendered) or crashes. One solution was suggested by Aftershock\_81:

```
COE 0 0

Player->SetPos x xpos
Player->SetPos x ypos
Player->SetPos x zpos

FixMe
```

where FixMe is meant to reload the destination cell to avoid the problem where SetPos does not force the cell to load correctly.

I only ever managed to get Aftershock's method to work via a local script, when I tried using a global script, everything worked, but I got an "Function greater than index count" error. A good example of using setpos and fixme can be found in Dongle's Ranger Tent mod, available on Planet Elder Scrolls.

If you use FixMe, you may also want to use SetPos again, as fixMe will move the player. It depends on how accurately you want to place the player.

### Example script, by Nigedo (based on the work of Aftershock\_81 and JOG)

Note: This script **must** be a local script. Attach it to an activator or door: It will not work as a global script!

```
Begin script_PlacePC
;Global Float Start_PCX
;Global Float Start_PCY
;Global Float Start_PCZ
Float xpos
Float ypos
Float zpos
Float timer
Short step

If ( OnActivate )
    Set step to 1
Endif

If ( step == 1 )
    FadeOut 0.1
    ToggleMenus
    Set step to 2
    Return
Elseif ( step == 2 )
    Set timer to ( timer + GetSecondsPassed )
    If ( timer > 0.2 )
        Set timer to 0
        Set step to 3
    Endif
    Return
Elseif ( step == 3 )
    Set step to 0
    Player->COE 0 0

    Set xpos to Start_PCX
    Set ypos to Start_PCY
    Set zpos to Start_PCZ

    ToggleMenus
```

```
MenuTest
MenuTest

Player->SetPos x xpos
Player->SetPos y ypos
Player->SetPos z zpos

FixMe

Player->SetPos x xpos
Player->SetPos y ypos
Player->SetPos z zpos

FadeIn 0.1
Endif
End
```

If it is not possible to simply attach the script to an activate-able item, one alternative is to have another script (global or local) place an item in the player's inventory temporarily, with a teleport script attached to the inventory item (thanks to Kaos\_nyrb and Nigedo).

## Interaction between mods

(Thanks to Emma, TheOtherFelix and Ragnar\_GD for much of this information.)

### Some cautions

- If you want your mod to include interaction with someone else's mod, the modder in question should be contacted first, and permissions asked.
- An update to the mod you wish to interact with may destroy compatibility if certain things are changed (another reason to ask permission first!).
- If the mod you wish to interact with is removed from the load list, this may cause problems (depending on what you want to do).
- Be very careful! Always test thoroughly to make sure you haven't broken anything.

### Global variables

Global variables are not unique to a single mod: if another mod contains a variable of the same name, the last mod to load will overwrite the other, and both mods can set the global from script (*Note by GBG: – one more reason why people should try to use unique names wherever possible in their mods – don't name your global "check" at least call it (your initials)\_check, e.g. "YI\_check" – that avoids lots of problems and compatibility issues*). This can be used to facilitate interaction between mods: by implementing a global of the same type with the same default value in your own mod, it is possible to test for the value of that global in script or dialogue without interfering with the other mod (as long as you **never set the global**, and make sure you give it the same default value and type, it will have no effect on the other mod). When the other mod sets the global to some value other than the default, this can be detected.

Some modders have used this simply to "declare" a mod to others, by setting a global (usually from a startscript) so that it can be detected from script or dialogue. In other cases the interaction is more complex: for example, since a global can be tested from dialogue, and dialogue resultbox scripts aren't compiled until the line is "said" in game, a global can be used as a dialogue filter for a line whose resultbox script can only be run while the second mod is loaded (e.g. starting a script or referencing an ID that doesn't exist in your mod). Another common use is in "follow globals" for companions, which may be tested in dialogue or script to detect whether an NPC from a different mod is currently in AIFollow mode.

### Local variables

Emma and TheOtherFelix introduced this ingenious idea with "emmasnpcid": this local variable is declared and set to a unique value for each of Emma's companions, and can then be tested for those values in dialogue. This makes it possible to identify the companion the player is currently in dialogue with, thus opening up a great range of possibilities.

A simpler use of local variables in dialogue is to make dialogue from your mod *unavailable* to NPCs with that local variable, in cases where other modders might want to prevent their NPCs being affected by a particular mod.

### Journal entries

It is possible to test journal entries from other mods, in much the same way as for globals. To do this, create a dummy journal topic with the same ID as the original (leave the topic blank). You can then test the journal index in your own mod. This does not interfere with the journal entries added by the original mod.

## ***Safely starting global scripts- avoiding the main script***

With the expansion, this issue is a problem no more – just add the script you want to start running to the list of "Start Scripts" by selecting Edit Start Scripts from the Gameplay menu. This script will now be automatically started when a game is loaded, just like the main script. For those without Tribunal/Bloodmoon:

Many modders like to add a line to the main script (the only script that is started by default when a new game is started, and always runs) to make sure a global script is started that is essential to their mod. For example:

```
StartScript "My_Script"
```

While this works, it might easily cause conflicts with mods that used the same approach – because only the changes of the last loaded plugin will be active in the game. There are some good alternatives to using the main script. Putting an (invisible) activator in the Seyda Neen census office with the following attached script will make sure the script is started during character creation (assuming no alternative char gen mods are used alongside your mod):

```
Begin Script_launcher
If ( ScriptRunning, My_Script == 0 )
    StartScript, My_Script
    ; MessageBox "MyScript was activated" ; tell the player if you like
endif
End
```

To make sure a global script is started in an already running game you can use a similar method, and place activators in commonly visited cells. If you have one in Balmora, Vivec, Sadrith Mora, Dagon Fel, and Caldera and maybe the PC strongholds, I am sure it won't take long until the script is running. You can also use an object that is required for your mod anyway to start it. E.g. for Indestructibles excellent Bank mod, I made myself a version that attaches the above script to the banner of the bank and starts the "interest" script. That makes sure that that script is running before the PC ever sets foot inside the bank.

## **Use sound to detect events**

This to me was a very smart idea (thanks to BalorNG), so it bears mentioning again here, although it's also described in the sound section above. You can use the `GetSoundPlaying` function to determine certain events in the game that would otherwise not be accessible. Just take a look at the sounds in the Gameplay/sounds menu, and it may give you some ideas: determine if someone is falling, determine whether a certain monster is near, determine a hit with a weapon etc.

Here is some more info on this (thanks Horatio):

*GetSoundPlaying is a very powerful command that can be used to detect when the PC ( and I'm assuming other Actors ) is doing a certain action like casting a spell or swinging a weapon. I used it in my spellcasting mod to determine when the PC is casting a spell and what school of magic the spell is in. the format is as follows:*

```
if ( player->GetSoundPlaying, "Sound ID" == 1 )
;do something cool here
endif
```

*Look in the sounds menu in the TESCS to find which Sound ID corresponds to a specific action. For instance "illusion cast" corresponds to the player casting an illusion based spell. You'll probably have to experiment a little. Note: for some reason the Sound ID "drink" causes an error, so no checking if the PC is drinking a potion.*

## **Large battles**

(by Horatio)

The easiest way to do a large battle between two groups of NPCs is to use the AI commands. Let's use an example of a bunch of imperial legionnaires, with whom the PC is aligned, versus a dark brotherhood gang. First set the 'fight' rating ( in the AI tab) of the DB NPCs to 100 so that they'll attack the PC on sight. Then, you'll need to set the AI of the legionnaires to:

```
AIFollow player 0 0 0 0
```

You can do this either with scripts attached to the legion NPCs or with an external script. The default behavior of `AIFollow` is to attack whatever is attacking the person they are following. So when the DB guys attack the PC, all the legionnaires will freak out and starting attacking them back. Presto instant giant melee.

I use a variation on this in the GIANTS mod to convince the guards to attack monsters that are actually NPCs ( vampires, shades, giants, gorgos, etc ).

## ***A guide to making rideable objects***

(By MadMax\_001)

MadMax shares his insights on how to make rideable objects (e.g. boats) here, especially which problems will arise and what he did to solve them. There is a lot of interesting info here that goes beyond just this specific application, though. You can look at his scripts in the "Fishing Academy" and the "Magic Carpet" Mods (but don't use them in your mod without his consent).

### **Selecting objects**

Practically all objects (statics/activators) can be used. However, selecting the right type of objects is paramount. It will make your scripting a lot easier later on. Now, what type of objects are suitable? Preference is given to small and minimum height ( thickness ). The other important factor is the center point which is also the point where your character will be standing on. This will also save you a lot of programming work later on. If the object center point is not what you want, you can fix this by importing the object to 3DS and move the axis to the point to where you plan your character to stand. I am not going to explain in details how to do it in 3DS, there should be quite a number of tutorials out there which teach you how to do it.

### **Creating/Deleting objects**

I am sure a lot of modders know that you can only move an object through the exterior cells within certain distances. This is because the game will only update and process objects/codes within a certain distance. When your character gets out of that parameter, the object will actually be frozen or to the player, the object has warped/disappeared into thin air. But, if you trace back to the original cell, the object will re-appear.

Now, to move objects throughout the entire exterior cells, the trick to use here is to create a new object. Everytime when you move to a new cell, the function "CellChanged" will become TRUE for one frame. This is the best time to replace the existing object with a new one. There is one BIG problem that I discovered here. NEVER create an object from an object script. It doesn't seems to work. So, what you need to do is to use a global script to create one instead. At the same time, do not forget to Delete the old object or you will have all these objects spreading in different cells which will cause you major problems later on. Always maintain one object at one time. Below is a simple example you can use:

```
;-----  
; Object script  
;-----  
if ( player->CellChanged == 1 )  
Startscript, "Create_obj_script" ; this is the global script  
set obj_count to ( obj_count - 1 ) ; global parameter to count how many object exist  
Disable  
SetDelete, 1  
endif
```

```
;-----  
; Global script  
;-----  
PlaceAtPC "objectname", 1, 0, 0 ; or you can use PlaceItem  
set obj_count to ( obj_count + 1 )  
Stopscript "create_obj_script"
```

Ideally, the above script should work like a charm but in reality it is going to cause you major problems. Deleting an object immediately after changing cell may sometimes cause CTD. This is especially true when you have a heavy area loading. To fix this problem, delay the deletion. Introduce a time delay ( I personally find the 1.5seconds to be OK so far ). Here's how the object script will looks like now.

```
if ( player->CellChanged == 1 )
Startscript, "Create_obj_script" ; this is the global script
Disable
set timer_flag to 1
endif

if ( timer_flag == 1 )
set timer to ( timer + GetSecondsPassed )
if ( timer > 1.5 )
set obj_count to ( obj_count - 1 )
SetDelete, 1
else
return ; stop all other code processing
endif
endif
```

That is not all. If you plan to move your object at very high speed. There is a possibility that the object may encounter another cell change during the 1.5 seconds delay. You must make sure that the object gets deleted before it gets out of the processing parameter or it will come back and haunt you later. Now the script looks like this.

```
if ( player->CellChanged == 1 )
if ( timer_flag == 1 )
SetDelete, 1
return
endif
Startscript, "Create_obj_script"
Disable
set timer_flag to 1
endif

if ( timer_flag == 1 )
set timer to ( timer + GetSecondsPassed )
if ( timer > 1.5 )
SetDelete, 1
else
return
endif
endif
```

## Falling off from objects

This is actually very simple. Remove the gravitational effect on the character. This can be achieved by either introducing a levitation or floating ability to the character (**Note** by GBG: SetPOS also works for this purpose). The former being the perfect solution but it will disable your ability to make use of detecting RUN and SNEAK buttons. The latter allows you to detect movements but at the expense of possible falloff. You must also be wondering why your character falls off in the first place. The fact is that everytime when you create an object, the object collision parameter are not updated. It will be updated though when you change cell. You will find out that you can actually clip through the object with your character. The good news is that there is a way to work around this. By Disable and then Enable the object again will automatically update this. To make sure that the object is always "SOLID", perform the Disable and Enable at least once in every frame in the object script.

## **Collision detection**

This is biggest headache of all. Objects will not collide with objects. Only character/NPC/creatures can collide with objects. In this term, objects means statics/activators and landmass. The only way to detect collision is when your character hit the object. That is why I mentioned earlier that if you select a big object to ride, you will see clipping until the moment your character hit something. If you can live with that, that's fine otherwise it looks pretty awkward. The simplest method to detect collision is to get your player coordinates and measure against the object coordinates ( the one you are riding on ). Although, this is not fully proven, using GetSquareRoot function in the object script can sometimes cause CTD. There are 2 planes that you need to take care of. For eg, the flying carpet uses detection for both vertical (z axis) and horizontal (x, y axis ) planes. You can refer to my script on how this is done. If somebody can come up with a better method, please do share it.

## **Savegame issue**

If you have not realised already, when you save your game while in motion, the object coordinates updated in the your savegame are the ones during the cell changed. In order to position the object correctly, it is always best to keep a global parameters of the object coordinates. When you first load the game, do a detection on the object existing coordinates vs its global coordinates. If there is a big discrepancy, set the object to the global coordinates. In this way, when you first load the game, the object will be in precisely the same position when you save it.

## Trigonometry script - fast sine and cosine

I asked JDGBOLT to share his latest, greatest trigonometry script with the readers of MSFD, and I am happy to present it here. Although it is very long, I believe this will be an invaluable resource for anyone doing scripts involving trigonometry, e.g. those involving directional movement. I have edited and commented the original script to make it easy to use for any modder. Please give JDGBOLT credit if you use this!

The script calculates the sine and cosine of three angles. These will usually be the angles of all three axes of an object, obtained with GetAngle. Store these angles in the global variables

```
float Z_input_variable  
float X_input_variable  
float _input_variable
```

You can do this e.g. from another script on the object you are moving. Then run the script:

Startscript jdtrigscript. The script in this version is self-terminating and will place the results in the following global variables:

```
Float Z_sin  
Float Z_cos  
Float X_sin  
Float X_cos  
Float Y_sin  
Float Y_cos
```

You need to create all of the global variables before you can compile this script. The script is very fast, and the results very precise. The results will be available after one frame.

(Extra thanks to JDGBOLT for sharing his script! By the way, this is the colored script output from MWEedit, see the tip on alternative editors.)

```
begin jdtrigscript  
; by JDGBOLT  
; edited and commented for MSFD by GhanBuriGhan  
; simultaneous calculation of sine and cosine for three angles (e.g. three different axes)  
  
; for intermediate results of sine and cosine  
float ax1  
float ax2  
float ay1  
float ay2  
  
; main angle variable  
float angle  
  
;temp storage for angle:  
float angle_source ;contains angle (loaded from global)  
float angle_temp  
float angle_temp2  
  
short angleshort ; short variable to take up angle - for tab values  
float angledec ; for decimal portion of angle  
  
short anglequad ;quadrant  
  
float axfinal ; for final result sine  
float ayfinal ; for final result cosine  
  
float angconvx1 ; for calculating decimal portion of sine and cosine  
float angconvy1  
float angconvx2  
float angconvy2  
  
short objnum ;counter to calculate all three angles  
  
;-----  
while ( objnum <= 3 ) ; this script calculates sine and cosine for all three axis in one frame  
;exporting results:  
if ( objnum == 1 )  
    set Z_sin to axfinal ; assign to global var. Z_sin to free axfinal for next axis  
    set Z_cos to ayfinal ; ditto  
endif  
if ( objnum == 2 )  
    set X_sin to axfinal ;ditto
```

```

    set X_cos to ayfinal ;ditto
endif
if ( objnum == 3 )
    set Y_sin to axfinal ;ditto
    set Y_cos to ayfinal ;ditto
    ;MessageBox "sine of Z: %.3f cosine of Z: %.3f", Z_sin, Z_cos
    ;MessageBox "sine of X: %.3f cosine of X: %.3f", X_sin, X_cos
    ;MessageBox "sine of Y: %.3f cosine of Y: %.3f", Y_sin, Y_cos
    Set objnum to 0
    StopScript jdrtrigscript
    Return
endif
;-----
set objnum to ( objnum + 1 ) ;count up to do all three angles one by one
;-----
; obtain the angles from global variables.
if ( objnum == 1 )
    set angle_source to Z_input_angle
endif
if ( objnum == 2 )
    set angle_source to X_input_angle
endif
if ( objnum == 3 )
    set angle_source to Y_input_angle
endif

;GetAngle returns values from -180 to +180, so we need to compensate for calculation to get
;values from 0 to 360°C:
if ( angle_source < 0 )
    set angle_source to ( 360 + angle_source )
endif

set angle to angle_source

;-----
;determine quadrant. This way we need to calculate sine and cosine only for 0-90°
if ( angle <= 90 )
    set angle_temp2 to angle
    set anglequad to 1
elseif ( angle <= 180 )
    set angle_temp2 to ( 180 - angle )
    set anglequad to 2
elseif ( angle <= 270 )
    set angle_temp2 to ( angle - 180 )
    set anglequad to 3
elseif ( angle <= 360 )
    set angle_temp2 to ( 360 - angle )
    set anglequad to 4
endif

; obtain the decimal part of the angle
set angleshort to angle_temp2
set angledec to ( angle_temp2 - angleshort ) ; the difference between the float and the short
is the decimal part
set angle_temp to angleshort
;-----
; These are pre-calculated values for sine and cosine:
if ( angle_temp <= 10 ) ; angles arranged in chunks of 10 for speed (and if block limits?)
    if ( angle_temp == 0 )
        set ax1 to 0.000000 ; sine
        set ay1 to 1.000000 ;cosine
        set ax2 to 0.017452 ; sine of angletemp+1
        set ay2 to 0.999847 ;cosine of angletemp+1
    elseif ( angle_temp == 1 )
        set ax1 to 0.017452
        set ay1 to 0.999847
        set ax2 to 0.034899
        set ay2 to 0.999390
    elseif ( angle_temp == 2 )
        set ax1 to 0.034899
        set ay1 to 0.999390
        set ax2 to 0.052333
        set ay2 to 0.998629
    elseif ( angle_temp == 3 )
        set ax1 to 0.052333
        set ay1 to 0.998629
        set ax2 to 0.069756
        set ay2 to 0.997564
    
```

```

elseif ( angle_temp == 4 )
    set ax1 to 0.069756
    set ay1 to 0.997564
    set ax2 to 0.087155
    set ay2 to 0.996194
elseif ( angle_temp == 5 )
    set ax1 to 0.087155
    set ay1 to 0.996194
    set ax2 to 0.104528
    set ay2 to 0.994521
elseif ( angle_temp == 6 )
    set ax1 to 0.104528
    set ay1 to 0.994521
    set ax2 to 0.121869
    set ay2 to 0.992546
elseif ( angle_temp == 7 )
    set ax1 to 0.121869
    set ay1 to 0.992546
    set ax2 to 0.139173
    set ay2 to 0.990268
elseif ( angle_temp == 8 )
    set ax1 to 0.139173
    set ay1 to 0.990268
    set ax2 to 0.156434
    set ay2 to 0.987688
elseif ( angle_temp == 9 )
    set ax1 to 0.156434
    set ay1 to 0.987688
    set ax2 to 0.173648
    set ay2 to 0.984807
elseif ( angle_temp == 10 )
    set ax1 to 0.173648
    set ay1 to 0.984807
    set ax2 to 0.190808
    set ay2 to 0.981627
endif
elseif ( angle_temp <= 20 ) ; and so on...
if ( angle == 11 )
set ax1 to 0.190808
set ay1 to 0.981627
set ax2 to 0.207911
set ay2 to 0.978147
elseif ( angle_temp == 12 )
set ax1 to 0.207911
set ay1 to 0.978147
set ax2 to 0.224951
set ay2 to 0.974370
elseif ( angle_temp == 13 )
set ax1 to 0.224951
set ay1 to 0.974370
set ax2 to 0.241921
set ay2 to 0.970295
elseif ( angle_temp == 14 )
set ax1 to 0.241921
set ay1 to 0.970295
set ax2 to 0.258819
set ay2 to 0.965925
elseif ( angle_temp == 15 )
set ax1 to 0.258819
set ay1 to 0.965925
set ax2 to 0.275637
set ay2 to 0.961261
elseif ( angle_temp == 16 )
set ax1 to 0.275637
set ay1 to 0.961261
set ax2 to 0.292371
set ay2 to 0.956304
elseif ( angle_temp == 17 )
set ax1 to 0.292371
set ay1 to 0.956304
set ax2 to 0.309016
set ay2 to 0.951056
elseif ( angle_temp == 18 )
set ax1 to 0.309016
set ay1 to 0.951056
set ax2 to 0.325568
set ay2 to 0.945518
elseif ( angle_temp == 19 )

```

```

set ax1 to 0.325568
set ay1 to 0.945518
set ax2 to 0.342020
set ay2 to 0.939692
elseif ( angle_temp == 20 )
set ax1 to 0.342020
set ay1 to 0.939692
set ax2 to 0.358367
set ay2 to 0.933580
endif
elseif ( angle_temp <= 30 )
if ( angle_temp == 21 )
set ax1 to 0.358367
set ay1 to 0.933580
set ax2 to 0.374606
set ay2 to 0.927183
elseif ( angle_temp == 22 )
set ax1 to 0.374606
set ay1 to 0.927183
set ax2 to 0.390731
set ay2 to 0.920504
elseif ( angle_temp == 23 )
set ax1 to 0.390731
set ay1 to 0.920504
set ax2 to 0.406736
set ay2 to 0.913545
elseif ( angle_temp == 24 )
set ax1 to 0.406736
set ay1 to 0.913545
set ax2 to 0.422618
set ay2 to 0.906307
elseif ( angle_temp == 25 )
set ax1 to 0.422618
set ay1 to 0.906307
set ax2 to 0.438371
set ay2 to 0.898794
elseif ( angle_temp == 26 )
set ax1 to 0.438371
set ay1 to 0.898794
set ax2 to 0.453990
set ay2 to 0.891006
elseif ( angle_temp == 27 )
set ax1 to 0.453990
set ay1 to 0.891006
set ax2 to 0.469471
set ay2 to 0.882947
elseif ( angle_temp == 28 )
set ax1 to 0.469471
set ay1 to 0.882947
set ax2 to 0.484809
set ay2 to 0.874619
elseif ( angle_temp == 29 )
set ax1 to 0.484809
set ay1 to 0.874619
set ax2 to 0.500000
set ay2 to 0.866025
elseif ( angle_temp == 30 )
set ax1 to 0.500000
set ay1 to 0.866025
set ax2 to 0.515038
set ay2 to 0.857167
endif
elseif ( angle_temp <= 40 )
if ( angle_temp == 31 )
set ax1 to 0.515038
set ay1 to 0.857167
set ax2 to 0.529919
set ay2 to 0.848048
elseif ( angle_temp == 32 )
set ax1 to 0.529919
set ay1 to 0.848048
set ax2 to 0.544639
set ay2 to 0.838670
elseif ( angle_temp == 33 )
set ax1 to 0.544639
set ay1 to 0.838670
set ax2 to 0.559192
set ay2 to 0.829037

```

```

elseif ( angle_temp == 34 )
set ax1 to 0.559192
set ay1 to 0.829037
set ax2 to 0.573576
set ay2 to 0.819152
elseif ( angle_temp == 35 )
set ax1 to 0.573576
set ay1 to 0.819152
set ax2 to 0.587785
set ay2 to 0.809016
elseif ( angle_temp == 36 )
set ax1 to 0.587785
set ay1 to 0.809016
set ax2 to 0.601815
set ay2 to 0.798635
elseif ( angle_temp == 37 )
set ax1 to 0.601815
set ay1 to 0.798635
set ax2 to 0.615661
set ay2 to 0.788010
elseif ( angle_temp == 38 )
set ax1 to 0.615661
set ay1 to 0.788010
set ax2 to 0.629320
set ay2 to 0.777145
elseif ( angle_temp == 39 )
set ax1 to 0.629320
set ay1 to 0.777145
set ax2 to 0.642787
set ay2 to 0.766044
elseif ( angle_temp == 40 )
set ax1 to 0.642787
set ay1 to 0.766044
set ax2 to 0.656059
set ay2 to 0.754709
endif
elseif ( angle_temp <= 50 )
if ( angle_temp == 41 )
set ax1 to 0.656059
set ay1 to 0.754709
set ax2 to 0.669130
set ay2 to 0.743144
elseif ( angle_temp == 42 )
set ax1 to 0.669130
set ay1 to 0.743144
set ax2 to 0.681998
set ay2 to 0.731353
elseif ( angle_temp == 43 )
set ax1 to 0.681998
set ay1 to 0.731353
set ax2 to 0.694658
set ay2 to 0.719339
elseif ( angle_temp == 44 )
set ax1 to 0.694658
set ay1 to 0.719339
set ax2 to 0.707106
set ay2 to 0.707106
elseif ( angle_temp == 45 )
set ax1 to 0.707106
set ay1 to 0.707106
set ax2 to 0.719339
set ay2 to 0.694658
elseif ( angle_temp == 46 )
set ax1 to 0.719339
set ay1 to 0.694658
set ax2 to 0.731353
set ay2 to 0.681998
elseif ( angle_temp == 47 )
set ax1 to 0.731353
set ay1 to 0.681998
set ax2 to 0.743144
set ay2 to 0.669130
elseif ( angle_temp == 48 )
set ax1 to 0.743144
set ay1 to 0.669130
set ax2 to 0.754709
set ay2 to 0.656059
elseif ( angle_temp == 49 )

```

```

set ax1 to 0.754709
set ay1 to 0.656059
set ax2 to 0.766044
set ay2 to 0.642787
elseif ( angle_temp == 50 )
set ax1 to 0.766044
set ay1 to 0.642787
set ax2 to 0.777145
set ay2 to 0.629320
endif
elseif ( angle_temp <= 60 )
if ( angle == 51 )
set ax1 to 0.777145
set ay1 to 0.629320
set ax2 to 0.788010
set ay2 to 0.615661
elseif ( angle_temp == 52 )
set ax1 to 0.788010
set ay1 to 0.615661
set ax2 to 0.798635
set ay2 to 0.601815
elseif ( angle_temp == 53 )
set ax1 to 0.798635
set ay1 to 0.601815
set ax2 to 0.809016
set ay2 to 0.587785
elseif ( angle_temp == 54 )
set ax1 to 0.809016
set ay1 to 0.587785
set ax2 to 0.819152
set ay2 to 0.573576
elseif ( angle_temp == 55 )
set ax1 to 0.819152
set ay1 to 0.573576
set ax2 to 0.829037
set ay2 to 0.559192
elseif ( angle_temp == 56 )
set ax1 to 0.829037
set ay1 to 0.559192
set ax2 to 0.838670
set ay2 to 0.544639
elseif ( angle_temp == 57 )
set ax1 to 0.838670
set ay1 to 0.544639
set ax2 to 0.848048
set ay2 to 0.529919
elseif ( angle_temp == 58 )
set ax1 to 0.848048
set ay1 to 0.529919
set ax2 to 0.857167
set ay2 to 0.515038
elseif ( angle_temp == 59 )
set ax1 to 0.857167
set ay1 to 0.515038
set ax2 to 0.866025
set ay2 to 0.500000
elseif ( angle_temp == 60 )
set ax1 to 0.866025
set ay1 to 0.500000
set ax2 to 0.874619
set ay2 to 0.484809
endif
elseif ( angle_temp <= 70 )
if ( angle_temp == 61 )
set ax1 to 0.874619
set ay1 to 0.484809
set ax2 to 0.882947
set ay2 to 0.469471
elseif ( angle_temp == 62 )
set ax1 to 0.882947
set ay1 to 0.469471
set ax2 to 0.891006
set ay2 to 0.453990
elseif ( angle_temp == 63 )
set ax1 to 0.891006
set ay1 to 0.453990
set ax2 to 0.898794
set ay2 to 0.438371

```

```

elseif ( angle_temp == 64 )
set ax1 to 0.898794
set ay1 to 0.438371
set ax2 to 0.906307
set ay2 to 0.422618
elseif ( angle_temp == 65 )
set ax1 to 0.906307
set ay1 to 0.422618
set ax2 to 0.913545
set ay2 to 0.406736
elseif ( angle_temp == 66 )
set ax1 to 0.913545
set ay1 to 0.406736
set ax2 to 0.920504
set ay2 to 0.390731
elseif ( angle_temp == 67 )
set ax1 to 0.920504
set ay1 to 0.390731
set ax2 to 0.927183
set ay2 to 0.374606
elseif ( angle_temp == 68 )
set ax1 to 0.927183
set ay1 to 0.374606
set ax2 to 0.933580
set ay2 to 0.358367
elseif ( angle_temp == 69 )
set ax1 to 0.933580
set ay1 to 0.358367
set ax2 to 0.939692
set ay2 to 0.342020
elseif ( angle_temp == 70 )
set ax1 to 0.939692
set ay1 to 0.342020
set ax2 to 0.945518
set ay2 to 0.325568
endif
elseif ( angle_temp <= 80 )
if ( angle_temp == 71 )
set ax1 to 0.945518
set ay1 to 0.325568
set ax2 to 0.951056
set ay2 to 0.309016
elseif ( angle_temp == 72 )
set ax1 to 0.951056
set ay1 to 0.309016
set ax2 to 0.956304
set ay2 to 0.292371
elseif ( angle_temp == 73 )
set ax1 to 0.956304
set ay1 to 0.292371
set ax2 to 0.961261
set ay2 to 0.275637
elseif ( angle_temp == 74 )
set ax1 to 0.961261
set ay1 to 0.275637
set ax2 to 0.965925
set ay2 to 0.258819
elseif ( angle_temp == 75 )
set ax1 to 0.965925
set ay1 to 0.258819
set ax2 to 0.970295
set ay2 to 0.241921
elseif ( angle_temp == 76 )
set ax1 to 0.970295
set ay1 to 0.241921
set ax2 to 0.974370
set ay2 to 0.224951
elseif ( angle_temp == 77 )
set ax1 to 0.974370
set ay1 to 0.224951
set ax2 to 0.978147
set ay2 to 0.207911
elseif ( angle_temp == 78 )
set ax1 to 0.978147
set ay1 to 0.207911
set ax2 to 0.981627
set ay2 to 0.190808
elseif ( angle_temp == 79 )

```

```

set ax1 to 0.981627
set ay1 to 0.190808
set ax2 to 0.984807
set ay2 to 0.173648
elseif ( angle_temp == 80 )
set ax1 to 0.984807
set ay1 to 0.173648
set ax2 to 0.987688
set ay2 to 0.156434
endif
elseif ( angle_temp <= 90 )
if ( angle_temp == 81 )
set ax1 to 0.987688
set ay1 to 0.156434
set ax2 to 0.990268
set ay2 to 0.139173
elseif ( angle_temp == 82 )
set ax1 to 0.990268
set ay1 to 0.139173
set ax2 to 0.992546
set ay2 to 0.121869
elseif ( angle_temp == 83 )
set ax1 to 0.992546
set ay1 to 0.121869
set ax2 to 0.994521
set ay2 to 0.104528
elseif ( angle_temp == 84 )
set ax1 to 0.994521
set ay1 to 0.104528
set ax2 to 0.996194
set ay2 to 0.087155
elseif ( angle_temp == 85 )
set ax1 to 0.996194
set ay1 to 0.087155
set ax2 to 0.997564
set ay2 to 0.069756
elseif ( angle_temp == 86 )
set ax1 to 0.997564
set ay1 to 0.069756
set ax2 to 0.998629
set ay2 to 0.052335
elseif ( angle_temp == 87 )
set ax1 to 0.998629
set ay1 to 0.052335
set ax2 to 0.999390
set ay2 to 0.034899
elseif ( angle_temp == 88 )
set ax1 to 0.999390
set ay1 to 0.034899
set ax2 to 0.999847
set ay2 to 0.017452
elseif ( angle_temp == 89 )
set ax1 to 0.999847
set ay1 to 0.017452
set ax2 to 1.000000
set ay2 to 0.000000
elseif ( angle_temp == 90 )
set ax1 to 1.000000
set ay1 to 0.000000
set ax2 to 0.999847
set ay2 to 0.017452
endif
endif

;Use simple linear extrapolation to approximate a decimal value for sine and cosine:
set angconvx1 to ( ax2 - ax1 ) ; calculate distance sin(angle) and sin(angle+1)
set angconvy1 to ( ay2 - ay1 ) ; same for cos
set angconvx2 to ( angconvx1 * angledec ) ;calculate a fraction by the decimal residue
set angconvy2 to ( angconvy1 * angledec ) ; same for cos
set axfinal to ( ax1 + angconvx2 ) ; set final result to result + fraction
set ayfinal to ( ay1 + angconvy2 ) ; set final result to result + fraction

;correct for quadrant:
if ( anglequad == 1 )
set axfinal to ( axfinal * 1 ) ; holds final sine
set ayfinal to ( ayfinal * 1 ) ; holds final cosine
elseif ( anglequad == 2 )
set axfinal to ( axfinal * 1 ) ; holds final sine

```

```

    set ayfinal to ( ayfinal * -1 ) ; holds final cosine
elseif ( anglequad == 3 )
    set axfinal to ( axfinal * -1 ) ; holds final sine
    set ayfinal to ( ayfinal * -1 ) ; holds final cosine
elseif ( anglequad == 4 )
    set axfinal to ( axfinal * -1 ) ; holds final sine
    set ayfinal to ( ayfinal * 1 ) ; holds final cosine
endif

endwhile ; loop for all three axis

;-----
end jdtrigscript

```

## Mannequins

These are popular with many people as they are a nice way to show off your collected armor – you find them in many house mods – this shows how it is done (many thanks to Stephen Kent aka Riiak Shi Nal for sharing the script). *This example is a "next generation" one that uses Tribunal functions for checking weapon/armor to prevent PC from moving mannequin while weapons (Riiak has not yet figured out how to get mannequins to wield weapons) and/or armor are present on mannequin. Also split into two separate scripts to support both male and female versions of the mannequin. These changes do not prevent PC from picking up the mannequin while there are still misc items on it, those items will be lost.*

I have added some extra comments in addition to Riiak's. The Mannequin is in reality a normal NPC with 0 health or corpse (health set to 0 in TES CS). For this version you simply activate it to give it items, and it will equip armor you give it.

```
Begin rsn_mannequin_f_script

short button
short questionState
short nEquipType
short nStillEquipped
float fDeleteTimer

SkipAnim ;GBG: This is essential, it makes the Mannequin, which is an NPC, stand still

if ( menumode == 1 )
return
endif

if ( GetDisabled == 1 )
; if mannequin has been disabled then we need to wait some time then delete this reference
Set fDeleteTimer to ( fDeleteTimer + GetSecondsPassed )
if ( fDeleteTimer > 5 )
SetDelete 1
endif
return
endif

if ( OnActivate == 0 )
if ( questionState == 0 )
return
endif
endif

if ( questionState == 0 )
MessageBox, "Armor Mannequin", "Move Mannequin", "Add/Remove Armor"
set questionState to 1
endif

if ( questionState == 1 )
set button to GetButtonPressed

if ( button == 0 )
set questionState to 10
elseif ( button == 1 )
set questionState to 0
Activate
endif
endif

if ( questionState == 10 )
; This section is split into two groups of nested ifs because of nested if limits of
; the scripting language.
Set nStillEquipped to 0
; Here we check to see if a weapon is equipped (put in mainly for if anyone can figure
; out how to get a mannequin to show a weapon)
Set nEquipType to ( GetWeaponType )
if ( nEquipType == -1 )
; Here we check to see if any armor is equipped (NOTE: there are 10 different possible
; pieces of armor so we need to check each individually)
Set nEquipType to ( GetArmorType 0 )
```



```

        set questionState to 0
; Disable the current mannequin
; and create a new one we wouldn't have to worry about losing items)
; GBG: if Mannequins are transported a lot
; a SetDelete function might be a good idea
        Disable
; This is the item that contains the script
; to generate a new mannequin when dropped.
; GBG: for a "female" mannequin this would be a different item
player->addItem, "_rsn_man_f_holder", 1
playSound "Item Misc Up"
elseif ( button == 1 )
; There are still items on the mannequin (either from checks or user response)
set questionState to 0
Activate
endif
endif

end

```

The following script goes on the item that's added to our inventory when we move the mannequin. When you drop it, a new mannequin is created at your feet, hence the necessary removal of armor. You will lose it all otherwise:

```

; Script split into two scripts to handle the two different mannequin genders.

Begin rsn_man_f_holder_script

short OnPCDrop
float fDeleteTimer

if ( GetDisabled == 1 )
; if holder has been disabled then we need to wait some time then delete this reference
Set fDeleteTimer to ( fDeleteTimer + GetSecondsPassed )
if ( fDeleteTimer > 5 )
SetDelete 1
endif
return
endif

if ( OnPCDrop == 1 )
Disable
; This is the NPC with 0 health that is really just a standing corpse.
PlaceAtPC, "_rsn_mannequin_female", 1, 0, 0
Set OnPCDrop to 0
endif
end

```

## ***Is she looking at me?***

Here is a beautiful script by Horatio that allows to detect whether an Actor looks towards the player:

```
Begin PCLookAtMe

float fPCX
float fPCY
float fPCAngle
float fdx
float fdy
float fRatio

short sPCLookAtMe

set sPCLookAtMe to 1

;you could probably also add a GetLOS check in here
; however i could never get GetLOS to work properly...although i didn't try very hard

;is PC really far away
if ( GetDistance, Player > 8000 )
    set sPCLookAtMe to 0
else

;yay trigonometry
;this basically does a rough calculation of the PCs direction relative
;to the Actor, it only uses 45 degree chunks, though

    set fPCX to ( player->GetPos, X )
    set fPCY to ( player->GetPos, Y )
    set fPCAngle to ( player->GetAngle, Z )

    set fdx to GetPos, X
    set fdy to GetPos, Y

    set fdx to ( fdx - fPCX )
    set fdy to ( fdy - fPCY )

    set fRatio to ( fdx / fdy )

    if ( fdx > 0 )
        if ( fdy > 0 )
            if ( fRatio > 1 )
                if ( fPCAngle < -45 )
                    set sPCLookAtMe to 0
                endif
            else
                if ( fPCAngle < -90 )
                    set sPCLookAtMe to 0
                endif
                if ( fPCAngle > 135 )
                    set sPCLookAtMe to 0
                endif
            endif
        else
            if ( fRatio < -1 )
                if ( fPCAngle < 0 )
                    if ( fPCAngle > -135 )
                        set sPCLookAtMe to 0
                    endif
                endif
            else
                if ( fPCAngle < 45 )
                    if ( fPCAngle > -90 )
                        set sPCLookAtMe to 0
                    endif
                endif
            endif
        endif
    else
endif
endif
```

```

        if ( fdy > 0 )
            if ( fRatio < -1 )
                if ( fPCAngle > 45 )
                    set SPCLookAtMe to 0
                endif
            else
                if ( fPCAngle > 90 )
                    set SPCLookAtMe to 0
                endif

                if ( fPCAngle < -135 )
                    set SPCLookAtMe to 0
                endif
            endif
        else
            if ( fRatio > 1 )
                if ( fPCAngle > 0 )
                    if ( fPCAngle < 135 )
                        set SPCLookAtMe to 0
                    endif
                endif
            else
                if ( fPCAngle > -35 )
                    if ( fPCAngle < 90 )
                        set SPCLookAtMe to 0
                    endif
                endif
            endif
        endif
    endif

endif

if ( SPCLookAtMe == 0 )
;do something while the PC is not looking
endif

End

```

## Cinematic sequence

The following is a very smart approach to do a cinematic sequence by gianluca (Morrowind Summit forums). It removes player control, places the player on an invisible "CollisionWall" object and then moves him (and thus the camera) around. You can't have cinematic sequences involving the PC, but it's still great.

```
If menumode==1
return
endif

if doOnce==0
"Collision wall2"->disable
"Collision wall3"->disable
"Collision wall4"->disable
set doOnce to 1
endif

if doOnce==1
"Collision wall1"->moveworld X 800
messagebox "moving"
if ( "Player"->getPos Z < 570 )
set doOnce to 2
set playxx to "Player"->getPos X
set playyy to "Player"->getPos Y
set playzz to "Player"->getPos Z
"Collision wall2"->enable
"Player"->position -114679 -4119 590 90
endif
endif

if doOnce==2
"Collision wall2"->moveworld X 800
messagebox "moving"
if ( "Player"->getPos Z < 570 )
set playxx to "Player"->getPos X
set playyy to "Player"->getPos Y
set playzz to "Player"->getPos Z
"Collision wall3"->enable
"Player"->position -112634 -4119 590 90
set doOnce to 3
endif
endif

if doOnce==3
"Collision wall3"->moveworld X 200
"Collision wall3"->moveworld Y -800
if ( "Player"->getPos Z < 570 )
set doOnce to 4
set playxx to "Player"->getPos X
set playyy to "Player"->getPos Y
set playzz to "Player"->getPos Z
"Collision wall4"->enable
"Player"->position -112126 -6150 590 90
endif
endif

if doOnce==4
"Collision wall4"->moveworld X 600
"Collision wall4"->moveworld Y -450
if ( "Player"->getPos Z < 570 )
set doOnce to 5
set playxx to "Player"->getPos X
set playyy to "Player"->getPos Y
set playzz to "Player"->getPos Z
endif
endif

if doOnce==5
stopscript ELDQ_visualforbattle
endif
end ELDQ_visualforbattle
```

# Troubleshooting

## General hints

- A good way to debug is to insert MessageBox commands at key points in your script.
- If you produce error messages and can't find the cause, try to remove suspect lines of codes one by one (using ; to mark them as comments) to pinpoint the error causing line.
- Pay attention to the error codes that the editor and the game give you, they usually let you pinpoint the source of the problem to some degree.
- When you make changes to a script, it's a very good idea to save the script **twice** afterwards, to make sure the compiler properly checks it for errors.  
For example, if you delete a variable declaration but forget to remove the variable from the script, the compiler may not report an error the first time you save the script. This also happens when you change the name of a script (in the 'Begin' statement), but leave the original name in a StopScript statement.  
Apparently, the compiler doesn't update its internal list of variables, script names, etc. until **after** it compiles the script. End result is that the compiler will use the stored variable declarations and script names from its previous pass through the script, rather than the current ones.  
This probably explains why you have to save a script at least once before its name (from the 'Begin' statement) will be recognized in a StopScript statement in the same script. -(DinkumThinkum)

## The Console

### Using the Console to check variables:

In the game you can use the console to check on the state of variables. Bring up the console window (standard key is ~ or whatever key is left of the "1" key if you are using a non-US keyboard layout) and type "sv" – this lists all the global variables with their values, then all running global scripts (including targeted scripts) with their variables. Now find an object that has a script running on it. Bring up the console again, left click on the object – the console window title will change. Type "sv" again – now the local variables for the script running on this object will be listed. To check a single variable, use "show var\_name", however, if trying to get a var from a object that doesn't have a script, it can sometimes crash the game.

### Using the Console to quickly test scripts:

The console can help you test your scripts. For objects you don't need to place them with the editor and then go there. Just write down the ID for your object, load any savegame and then, in the console type:

```
PlaceatPC "My_Object" 1,1,1
```

This will drop the object directly at your feet.

```
Player->AddItem "My_Object" 1
```

This will drop the item into your inventory

To go to a specific location use

```
coc "cell_name"
```

for interior cells or

```
coe -1, -7
```

for exterior cells (write down the cell coordinates in the editor). coc works for exterior cells too but since most exterior cells have non-unique names you may not land quite where you want. It can still be useful though, for example coc balmora will take you somewhere in Balmora.

tcl

Toggles collision – float through walls, visit difficult to reach places easily.

Tgm

Toggle God mode – test without worrying about some monster killing you.

Also, the invaluable console commands to run the game in "debug" mode allow one to view hard numbers about spell effects, chances to hit, etc (thanks to Wakim). Some of these are:

1) press the "~" key to call up the console.

2) type "tcs" into the console and hit return.

3) click (right or left click, can't recall) anywhere on the screen outside of the console box to continue game time while leaving the box displayed and active.

That's it. There are other variants on the "tcs" command, such as (but not limited to) "tks" and "tms" which all stand for "toggle xxxxxx statistics" where xxxxxx is: c = combat, m = magic, k = kill, or what have you.

## ***Error messages, malfunctions and common causes***

### **In the editor**

The editor usually indicates the line in which the problem was encountered, so often errors reported here are relatively easy to fix.

#### **"Mismatched If/else/endif starting on line..."**

One or several if statements are not closed. Use tabs to facilitate finding missing endifs. This also happens when wrong names are used for variables functions or other typing errors.

#### **"Function reference object "Foobject" not found"**

This error indicates that the object in question e.g.

```
Foobject->GetDistance Player
```

Does not exist. When you write a script and then create a new object to be thus referenced you will also get this error. Close the editor window and reopen it to register the object with the compiler.

#### **"Could not find variable or function "Foobject""**

Another syntax error, indicating you have used an undefined variable, function or a non-existent object.

#### **"Script command "foofunc" not found on line 3"**

A command / function was not recognized by the compiler. Usually results from typos.

#### **"You need to end a script with script End scriptname"**

Indicates that the compiler did not find the End command. This error can also appear when a previous error disrupts the compilation process.

#### **"Syntax error Line 20. Miss matched parenthesis"**

Indicates that you do not close all parentheses you opened or vice versa.

#### **"You need to enter a value on line 20"**

You have not supplied all the arguments needed by a specific function.

### **In game error messages:**

**"EXPRESSION in script..."**

Followed by

### **"RightEval ..."**

This error indicates that Variables are not declared. Most often this happens with game variables that are almost used like functions, e.g. OnPCEquip.

EXPRESSION in general appears mostly when variables have not been declared in the script.

### **"LeftEval"**

This error seems to come up when you have accidentally declared a Function as a variable. The following lines e.g. would produce this error:

```
short ScriptRunning
if (ScriptRunning "MyScript" == 1 )
```

Both of the above can also be caused by not having spaces at the proper places. Always leave a space between parentheses and function calls, variables etc.

If ( OnActivate == 1 ), not if (OnActivate==1). This only causes errors very rarely, but it sometimes does, trust me.

### **"Infix to Postfix" error**

Usually indicates bad syntax. Can be caused by bad set commands using the "fix" arrow:

```
set somevar to ActorID->GetHealth
```

should be changed to:

```
set somevar to ( ActorID->GetHealth )
```

An alternative is again a forgotten variable declaration of variable type functions, e.g. of OnPCEquip, etc. (Thanks Horatio and Ragnar\_GD)

### **AITravel command does not work**

A common cause is coordinates erroneously set too far away (e.g. by omitting a "-" or having a digit too much in there). I once had a very nasty one of having typed two "-" which looked like one just slightly longer than usual "-" in the editor.

### **Doubled Objects**

There are two reasons why you may get doubled items in your game

1. You changed your mod load order (or added a mod in the middle of the load order) and then continued on from a save. This is very easy to do accidentally by just saving a mod in the CS, which will move the mod in the load order.
2. You altered a object that had information about it stored in the save or Added/removed a object that is in the same cell as a object with information stored about it in a save.

The best way to prevent doubling when testing work in progress mods is to always test a mod on a save that doesn't have the mod loaded or even better a new game.

Wrye has notes on these in much greater detail  
<http://wrye.ufrealms.net/Wrye%20Matching.html>

**Script fails to start, or stops unexpectedly**

This can happen when a variable is not declared but the script was nevertheless successfully saved, for example if you deleted a variable declaration or changed the variable name after saving and didn't update it correctly in the script. It can also be caused by the same thing in another (different) script.

**Crash to desktop when executing the script**

There are unfortunately a great number of possible causes. Many are connected with not having "do once" conditions (e.g. calling certain functions every frame). Other known problems: Removing objects with running scripts from within their own script. Using Equip on anything but potions (fixed with Tribunal). Casting targeted spells from an activator (fixed with Tribunal). Using AIActivate on teleport doors that lead outside of the active cell. Trying to use PlaceItem with the same Object ID the script is running on. Using SetDelete on a non-disabled object or an object with a targeted script running on it.

**Crash to Desktop (CTD) upon loading the plugin**

One reported reason is overlong calculations: there appears to be an issue with very long additions (e.g. adding up more than 20 variables in one line of code) that causes a mod to CTD upon loading. If this happens, split the calculations to several lines.

## Appendix

### *New functions that come with TRIBUNAL*



The Tribunal expansion for Morrowind introduced a number of new functions and fixes and extends several of the old ones. Note that both you and the user of the mod must have Tribunal (or Bloodmoon, or GOTY edition) installed to make use of these, so make sure you mark your mod accordingly. The functions are now sorted into the main document, but this list may serve as an easy reference to see which functions are Tribunal functions. (Many thanks to Bethesda's Mike Lipari for sharing the information on these functions with the community.)

### **Changes / fixes to Morrowind scripting:**

- SetPos now accepts variables (float) as arguments.
- SetAngle now accepts variables (float) as arguments.
- Equip now works as intended and forces NPC to equip (put on) armor and clothing.
- AIActivate was apparently fixed.

### **Index of new Tribunal Script Functions:**

|                       |                       |
|-----------------------|-----------------------|
| AddToLevCreature      | ModWaterLevel         |
| AddToLevItem          | PlaceItem             |
| ClearForceJump        | RemoveFromLevCreature |
| ClearForceMoveJump    | RemoveFromLevItem     |
| ClearForceRun         | SetDelete             |
| DaysPassed (variable) | SetJournalIndex (?)   |
| DisableLevitation     | SetScale              |
| EnableLevitation      | SetWaterLevel         |
| ExplodeSpell          |                       |
| ForceJump             |                       |
| ForceMoveJump         |                       |
| ForceRun              |                       |
| GetArmorType          |                       |
| GetCollidingActor     |                       |
| GetCollidingPC        |                       |
| GetForceJump          |                       |
| GetForceMoveJump      |                       |
| GetForceRun           |                       |
| GetPCJumping          |                       |
| GetPCRunning          |                       |
| GetPCSneaking         |                       |
| GetScale              |                       |
| GetSpellReadied       |                       |
| GetSquareRoot         |                       |
| GetWaterLevel         |                       |
| GetWeaponDrawn        |                       |
| GetWeaponType         |                       |
| HasItemEquipped       |                       |
| HurtCollidingActor    |                       |
| ModScale              |                       |

## ***New functions that come with BLOODMOON***



The second expansion for Morrowind, Bloodmoon, introduces a few new functions. Note that both you and the user of the mod must have Bloodmoon installed to make use of these, so make sure you mark your mod accordingly.

### **Index of new Bloodmoon Script Functions and variables:**

AllowWereWolfForceGreeting

BecomeWerewolf

GetPCInJail

GetPCTraveling

GetWerewolfKills

IsWerewolf

PCKnownWerewolf

PlaceAtMe

SetWerewolfAcrobatics

TurnMoonRed

TurnMoonWhite

UndoWerewolf

## ***Previously undocumented functions***

These functions were not documented in the original helpfile. I have explained most of them in this guide, but for a few the uses are still unknown to me. I thought it might be useful to have as a list for an easy overview. The list was put together by someone who Hex-edited the TES-CS .exe, so it's likely to be complete.

(thanks to Soralis for the list and XPCagey and others who helped to put it together):

### **Undocumented in Helpfile:**

```
PayFineThief
EnableStatReviewMenu
GetFactionReaction
ShowMap
EnableBirthMenu
EnableClassMenu
EnableRaceMenu
EnableNameMenu
RemoveEffects
EnableMagicMenu
EnableMapMenu
EnableInventoryMenu
EnableStatsMenu
GetInterior
GetLineOfSight (alias for GetLOS)
GetWindSpeed

GetCurrentTime
ResetActors
OutputRefInfo
MenuTest
```

### **Functions which are spelled differently than documented (XPCagey):**

```
getHealthRatio -> getHealthGetRatio
getInvisible -> getInvisibile
setInvisible -> setInvisibile
modInvisible -> modInvisibile
getSecundusPhase -> getSecundaPhase
(actual syntax on the right)
```

### **And those which don't work:**

```
getPlayerViewSwitch -> (can't be used)
OnRepair -> not in string tables
UsedOnMe -> not in string tables
name -> not in string tables
```

Also be aware that most console commands can be compiled as functions by the Construction Set (see list below). These are also to a large part not listed by the helpfile.

## Variable-type functions:

For easy reference, the following list shows all those variable/function type hybrids that you **need to declare** as variables if you use them in a script.

## Local variables that get set by the game:

Short OnPCEquip  
Short OnPCAdd  
Short OnPCRepair  
Short OnPCSoulGemUse  
Short OnPCHitMe  
Float minimumProfit (Tribunal)

## Local variables that you can set as a flag:

Short Companion (Tribunal)  
Short StayOutside (Tribunal)  
Short PCSkipEquip

## Special Globals

Some globals hold special significance that you can take advantage of for your own scripting. Since they are globals, you do not need to declare them.

|                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Short NPCVoiceDistance (750) | Used as a distance when Following NPCs call after you to wait for them (e.g. see DandsaScript)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Float GameHour               | Holds the current hour of the day (0-23)<br>Note, this is a float, so 10:30 would be represented as 10.5.<br>See the section on float variables for more information                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Short Day                    | Holds the current day of the month (1-30)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Short Month                  | Holds the current Month of the year (0-11)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Short Year (427)             | Holds the current year                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Float TimeScale (30)         | Sets the ratio of real-time/game-time                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Short Random100              | Is randomly set between 0-100 (set by main script when out of menumode)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Short PCRace                 | Contains the player's Race<br>(1=Argonian, 2=Breton, 3=Dark Elf, 4=High Elf, 5=Imperial, 6= Khajiit, 7=Nord, 8=Orc, 9=Redguard, 10=Woodelf)<br>If the player is using a custom race, PCRace will remain 0 by default. Note that it is possible to add your own racecheck script for custom races, and set PCRace to a custom value, but be warned: if two mods do this, you may have conflicts. As this variable is only used in dialog, it is not required that you set it to a different number for every custom race you make. If you made a new race with dialog, instead of using this it would be far better to make your own global variable as it would be less likely to conflict with other mods. |
| Short PCVampire              | Vampire status: 0=Normal, 1=Vampire, -1= cured                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Short VampClan               | If the PC becomes a vampire, this indicates his clan.<br>1=Aundae, 2=Berne, 3=Quarra                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Short DaysPassed             | Contains the number of days since the game started.<br>Requires an explicit declaration, which is present in Tribunal but must otherwise be added (not present in                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

|                                |                                                                                                                      |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------|
|                                | Bloodmoon or Morrowind).                                                                                             |
| Short PCWerewolf               | Werewolf status: 0=Normal, 1=Werewolf, -1=Cured                                                                      |
| Float WerewolfClawMult (25.00) | Increased during the Werewolf quests to make your claw attack more powerful.                                         |
| Short PCHasGoldDiscount        | Used in dialogue. Gets set to 1 if player has enough gold to pay thieves guild discount on the "price on your head". |
| Short PCHasTurnIn              | Used in dialogue. Gets set to 1 if player has enough gold to pay the reduced fee for turning yourself in.            |
| Short PCHasCrimeGold           | Used in dialogue. Gets set to 1 if player has enough gold to pay the crime fee.                                      |
| Short CrimeGoldTurnIn          | Gold needed for reduced crime fee                                                                                    |
| CrimeGoldDiscount              | Gold needed for thieves guild discount crime fee                                                                     |

### **Game units:**

1 game unit = 0.56 inches  
50 = 28 inches  
500 = 23.3 feet  
5000 = 233.3 feet  
8192 = 385 feet = 1 game cell

1 game unit = 1.42 cm  
100 game units = 142 cm = 1.42 meters  
1000 game units = 14.2 meters  
8192 game units = 116.33 meters = 1 exterior cell

The island of Morrowind itself is 5.00 km north to south and 4.65 km east to west.

(thanks to Iudas for this information)

### **Derived attribute calculations:**

*(Thanks to DinkumThinkum for this info.)*

The following is based on how they're calculated for player characters; checking in the Construction Set, NPC derived attributes appear to use basically the same system (if auto-calculate is enabled).

**Maximum Fatigue:** equals the total of current Strength, current Endurance, current Willpower, and current Agility.

**Maximum Magicka:** current Intelligence times the character's magicka multiplier (varies with race and birthsign). For NPCs, the multiplier apparently is fixed at two (default value of fNPCbaseMagickaMult: see Game Settings); race doesn't seem to affect it.

NOTE: this is based on the Maximum Magicka shown in the editor. Racial abilities that fortify Maximum Magicka are listed in the 'spells' section of the NPC's properties page, so that may be figured into the maximum value used during game play; haven't tested this.

**Maximum Encumbrance:** five times the character's current Strength.

**Maximum Health (player):** When you create your character, their starting maximum Health (at first level) is one half the total of their Strength and Endurance. After that, each time your character goes up a level their maximum Health is increased by 10% of their Endurance (value set by fLevelUpHealthEndMult, see Game Settings). This is based on the character's base natural Endurance; magical modifications to Endurance are ignored.

Strength is only used to calculate the player character's starting value for maximum Health; it has no further affect on Health once the character has been created. Also, Endurance affects the amount added to the character's maximum Health each level; there's no recalculation of the amount added in previous levels.

**Maximum Health (NPCs):** A number of formulas for NPC health have been proposed, but I (melian) have yet to find one that works for more than one NPC. At base level, NPC Health appears to be auto-calculated in the same way as the player's: (STR + END)/2. After that it gets tricky, and most formulas fail quite dramatically at higher levels.

For *some* NPCs: The calculation is based on the character's Strength and Endurance, and Health increase factor per level is constant within three brackets: before STR has maxed (largest increase), after STR but before END has maxed (smaller increase), and after both have maxed (constant smaller increase from then on). It's usually easy enough to calculate a single NPC's Health at any given level by noting Health, Strength and Endurance at various levels and working out the increase factor per level in each of the three brackets (remembering that values are really floats: averaging a larger value will get a more accurate increase factor). This has worked for most NPCs I've used it on, whose Strength maxed well before Endurance did. I'm not sure of the progression for NPCs whose Endurance maxes first, and race may have something to do with it as well.

## Magic Effect List

For the function GetEffect, you need the ID-string itself (e.g. GetEffect sEffectWaterBreathing). For the RemoveEffects function, the effect number is used (e.g. RemoveEffects, 0)

|                                       |                                       |
|---------------------------------------|---------------------------------------|
| 0 => sEffectWaterBreathing            | 69 => sEffectCureCommonDisease        |
| 1 => sEffectSwiftSwim                 | 70 => sEffectCureBlightDisease        |
| 2 => sEffectWaterWalking              | 71 => sEffectCureCorprusDisease       |
| 3 => sEffectShield                    | 72 => sEffectCurePoison               |
| 4 => sEffectFireShield                | 73 => sEffectCureParalyzation         |
| 5 => sEffectLightningShield           | 74 => sEffectRestoreAttribute         |
| 6 => sEffectFrostShield               | 75 => sEffectRestoreHealth            |
| 7 => sEffectBurden                    | 76 => sEffectRestoreSpellPoints       |
| 8 => sEffectFeather                   | 77 => sEffectRestoreFatigue           |
| 9 => sEffectJump                      | 78 => sEffectRestoreSkill             |
| 10 => sEffectLevitate                 | 79 => sEffectFortifyAttribute         |
| 11 => sEffectSlowFall                 | 80 => sEffectFortifyHealth            |
| 12 => sEffectLock                     | 81 => sEffectFortifySpellpoints       |
| 13 => sEffectOpen                     | 82 => sEffectFortifyFatigue           |
| 14 => sEffectFireDamage               | 83 => sEffectFortifySkill             |
| 15 => sEffectShockDamage              | 84 => sEffectFortifyMagickaMultiplier |
| 16 => sEffectFrostDamage              | 85 => sEffectAbsorbAttribute          |
| 17 => sEffectDrainAttribute           | 86 => sEffectAbsorbHealth             |
| 18 => sEffectDrainHealth              | 87 => sEffectAbsorbSpellPoints        |
| 19 => sEffectDrainSpellpoints         | 88 => sEffectAbsorbFatigue            |
| 20 => sEffectDrainFatigue             | 89 => sEffectAbsorbSkill              |
| 21 => sEffectDrainSkill               | 90 => sEffectResistFire               |
| 22 => sEffectDamageAttribute          | 91 => sEffectResistFrost              |
| 23 => sEffectDamageHealth             | 92 => sEffectResistShock              |
| 24 => sEffectDamageMagicka            | 93 => sEffectResistMagicka            |
| 25 => sEffectDamageFatigue            | 94 => sEffectResistCommonDisease      |
| 26 => sEffectDamageSkill              | 95 => sEffectResistBlightDisease      |
| 27 => sEffectPoison                   | 96 => sEffectResistCorprusDisease     |
| 28 => sEffectWeaknessToFire           | 97 => sEffectResistPoison             |
| 29 => sEffectWeaknessToFrost          | 98 => sEffectResistNormalWeapons      |
| 30 => sEffectWeaknessToShock          | 99 => sEffectResistParalysis          |
| 31 => sEffectWeaknessToMagicka        | 100 => sEffectRemoveCurse             |
| 32 => sEffectWeaknessToCommonDisease  | 101 => sEffectTurnUndead              |
| 33 => sEffectWeaknessToBlightDisease  | 102 => sEffectSummonScamp             |
| 34 => sEffectWeaknessToCorprusDisease | 103 => sEffectSummonClannfear         |
| 35 => sEffectWeaknessToPoison         | 104 => sEffectSummonDaedroth          |
| 36 => sEffectWeaknessToNormalWeapons  | 105 => sEffectSummonDremora           |
| 37 => sEffectDisintegrateWeapon       | 106 => sEffectSummonAncestralGhost    |
| 38 => sEffectDisintegrateArmor        | 107 => sEffectSummonSkeletalMinion    |
| 39 => sEffectInvisibility             | 108 => sEffectSummonLeastBonewalker   |
| 40 => sEffectChameleon                | 109 => sEffectSummonGreaterBonewalker |
| 41 => sEffectLight                    | 110 => sEffectSummonBonelord          |
| 42 => sEffectSanctuary                | 111 => sEffectSummonWingedTwilight    |
| 43 => sEffectNightEye                 | 112 => sEffectSummonHunger            |
| 44 => sEffectCharm                    | 113 => sEffectSummonGoldensaint       |
| 45 => sEffectParalyze                 | 114 => sEffectSummonFlameAtronach     |
| 46 => sEffectSilence                  | 115 => sEffectSummonFrostAtronach     |
| 47 => sEffectBlind                    | 116 => sEffectSummonStormAtronach     |
| 48 => sEffectSound                    | 117 => sEffectFortifyAttackBonus      |
| 49 => sEffectCalmHumanoid             | 118 => sEffectCommandCreatures        |
| 50 => sEffectCalmCreature             | 119 => sEffectCommandHumanoids        |
| 51 => sEffectFrenzyHumanoid           | 120 => sEffectBoundDagger             |
| 52 => sEffectFrenzyCreature           | 121 => sEffectBoundLongsword          |
| 53 => sEffectDemoralizeHumanoid       | 122 => sEffectBoundMace               |
| 54 => sEffectDemoralizeCreature       | 123 => sEffectBoundBattleAxe          |
| 55 => sEffectRallyHumanoid            | 124 => sEffectBoundSpear              |
| 56 => sEffectRallyCreature            | 125 => sEffectBoundLongbow            |
| 57 => sEffectDispel                   | 126 => sEffectExtraSpell              |
| 58 => sEffectSoultrap                 | 127 => sEffectBoundCuirass            |
| 59 => sEffectTelekinesis              | 128 => sEffectBoundHelm               |
| 60 => sEffectMark                     | 129 => sEffectBoundBoots              |
| 61 => sEffectRecall                   | 130 => sEffectBoundShield             |
| 62 => sEffectDivineIntervention       | 131 => sEffectBoundGloves             |
| 63 => sEffectAlmsiviIntervention      | 132 => sEffectCorpus                  |
| 64 => sEffectDetectAnimal             | 133 => sEffectVampirism               |
| 65 => sEffectDetectEnchantment        | 134 => sEffectSummonCenturionSphere   |
| 66 => sEffectDetectKey                | 135 => sEffectSunDamage               |
| 67 => sEffectSpellAbsorption          | 136 => sEffectStuntedMagicka          |
| 68 => sEffectReflect                  |                                       |

## List of console commands

Console (in game only commands). Most are useful for debugging / testing only, but some of these (marked with \*) CAN be used in scripts. For functions that produce an output to console: you can continue playing with the console open by rightclicking outside of the console window.

| Command                                                                             | short | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------------------------------------------------------------------------|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| *CenterOnCell, " <i>Cell_ID</i> "                                                   | COC   | Places the PC in the named cell. Very useful for testing mods.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| *CenterOnExterior, X, Y                                                             | COE   | Places the PC in the exterior cell grid.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| CreateMaps<br>" <i>Filename.esp</i> "                                               |       | Creates map image file depending on the Create Maps Enable value in the Morrowind.INI file. If it is 1 (XBox), the file FILENAME.ESP.MAP is created in the DataFiles path with the map data (unknown format). If the value is 2 (Exterior Cell Maps) and you have created a directory /Maps/ in the main Morrowind game directory, this command will create a 256x256 high color bitmap of each exterior cell in the game. This command takes a long while even on fast computers as each cell in the game is loaded.<br>Note that CreateMaps won't generate an image of the cell in which the player is currently located. So if you're standing in your mod's landscape, all the cells around it are processed just fine, but you'll be missing the current cell. To work around this, move to a cell you don't care about before starting the CreateMaps process. You might also be able to move to an interior cell, but this is untested. - (Klinn) |
|                                                                                     | BC    | Beta Comment: Edit the morrowind.ini file to give a filename in the beta comment line:<br>Beta Comment File=BetaComment.txt<br>Then you can use the BC Command to make a note about an object in the game. You open the console and click something, then type your comment, like this:<br>BC "Root not attached well."<br>And in BetaComment.txt you get:<br>6/20/2004 (21:02) Morrowind.esm 5/8/2003 (21:07) Paul ex_t_root_03 Tel Vos (10,14) 85078 118468 4111 "Root not attached well."<br>The time I made the comment, the file the object is from, the modification time of that file, my name (windows log-in), the cell, the X, Y and Z Position of the object, and of course, the comment (forum info / ManaUser).                                                                                                                                                                                                                             |
| FillJournal                                                                         |       | add all entries to journal, takes a long time                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| *FillMap                                                                            |       | show all the towns / uniquely named cells on the full map. Takes a few seconds. Not recommended for scripts.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| *FixMe                                                                              |       | Jump 128 units away from where I am now. Good to get "unstuck"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| GetFactionReaction,<br>" <i>factionID</i> ", " <i>factionID</i> ",<br><i>amount</i> |       | The faction ID's are not optional, works in Console window only. Not sure about the actual meaning of the output.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Help                                                                                |       | Lists, and shows shorthand for most commands                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| MoveOneToOne                                                                        | MOTO  | This command changes the speed at which the PC actor (and presumably other actors) RUN. With MOTO active your walking and running speeds are the same, and the same animation will play for either walking or running.(IndigoRage)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| ObjectReferenceInfo                                                                 | ORI   | Lists info about the selected object, such as the cell it's in and the esm/esp file it originates from. Handy when identifying which mod added a certain item.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| OutputObjCounts                                                                     |       | Counts all objects in various categories. Output is written to the console window                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| OutputRefCounts                                                                     |       | Counts all references in various categories. Output is written to the console window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|                                                                                     | PT    | "Purge textures" makes the engine reload all textures. If you play in windows mode you can use this to test textures in-game, while editing them in another application.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Show <i>global_var</i>                                                              |       | Writes the value of the specified global variable to the console                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| ShowVars                                                                            | SV    | Lists global variables and variables in global scripts, or local variables if you click on an object with a local script first. Output to console.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| StopCellTest                                                                        | SCT   | Stops the cell test, player remains in currently loaded cell                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

|                      |      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------------|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TestCells            |      | Loads all cells in alphabetical order (On testing it seemed to be only interiors - Not sure about difference to TestInteriorCells)<br>Note: TestCells, TestInteriorCells and TestThreadCells is a bit buggy. Even if you load a savegame while it's in process, it will continue until the last cell is reached.                                                                                                                                                                                                                                       |
| TestInteriorCells    |      | Loads all interior cells in alphabetical order                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| TestThreadCells      |      | Loads all exterior cells, then all interior cells. Exterior cells load in the following order:<br>[biggest x number], [biggest y number]<br>[biggest x number], [biggest y number - 1]<br>When the "y" number reaches the last value this follows<br>[biggest x number - 3 _OR_ - 1], [biggest y number]<br>[biggest x number - 3 _OR_ - 1], [biggest y number - 1]<br>etc.<br>Interior cells load in alphabetical order.<br>The command pauses in menumode, but not exactly when the menumode is first triggered. It waits several frames. -(unknown) |
| TestModels           | T3D  | Test all objects and reports missing .nif files                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| *ToggleAI            | TA   | Stops AI, including combat. Useful in testing a cell without being attacked by nasties                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| ToggleBorders        | TB   | Shows borders of exterior cells                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| ToggleCombatStats    | TCS  | Allows you to monitor combat statistics in realtime. Enable this, then right-click outside the console window to continue playing with the console window open.<br>Note: When I tested, the output was also written to log.txt - not sure if this is by default.                                                                                                                                                                                                                                                                                       |
| *ToggleCollision     | TCL  | Turns collision on and off. Lets you float through walls. Can make Actors drop through floors, or float                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| ToggleCollisionBoxes | TCB  | Shows the collision boxes of all models                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| ToggleCollisionGrid  | TCG  | Outputs a matrix to the screen that probably indicates the current collision situation - I couldn't interpret it. Slows the game to a crawl                                                                                                                                                                                                                                                                                                                                                                                                            |
| ToggleDebugText      | TDT  | Displays some debugging info on the screen: Players animation groups, speed, heading, position, FPS, delta movement per frame, and some I could not identify                                                                                                                                                                                                                                                                                                                                                                                           |
| ToggleDialogueStats  | TDS  | Outputs result of persuasion attempts (and maybe other dialogue results?) to the console                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| *ToggleFogOfWar      | TFOW | Lets you see all of the local map                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| ToggleFullHelp       | TFH  | Shows you ownership and script on mouseover while in console mode or in the info box during normal play. Also displays info on how the game fills leveled lists (upon entering a cell or opening a container) in the console.                                                                                                                                                                                                                                                                                                                          |
| *ToggleGodMode       | TGM  | Makes you invulnerable.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| ToggleGrid           | TG   | Displays the current cell's coordinates, and a grid of the currently loaded cells on screen, indicating the activities of morrowinds "thread loading" (caching of cells)                                                                                                                                                                                                                                                                                                                                                                               |
| ToggleKillStats      | TKS  | When an actor is killed the name of the kill, the total number of kills, and (with Bloodmoon) werewolf kills, is displayed in the console                                                                                                                                                                                                                                                                                                                                                                                                              |
| ToggleLights         |      | Unknown, I did not get any effect or output from this.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| ToggleLoadFade       |      | Unknown. From the name it would seem to refer to the screen fading in and out on loading of a game or teleporting, but I could not detect a difference.                                                                                                                                                                                                                                                                                                                                                                                                |
| ToggleMagicStats     | TMS  | Displays info on active spell effects in the console. Gives effect #, spell name, and statistical info                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| *ToggleMenus         | TM   | Disables all menus (including the main save/load/options menu!). Menus stay invisible until console is brought up again (press console key twice). Not recommended for use in scripts - If the player uses the console, your script may end up disabling the menus completely!                                                                                                                                                                                                                                                                         |
| ToggleScripts        |      | Presumably stops script processing                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| ToggleScriptOutput   | TSO  | Unknown, I did not get any effect or output from this.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| ToggleStats          | TST  | Activates all stat debugging modes (ToggleCombatStats, ToggleMagicStats, ToggleDialogueStats, ToggleKillStats)                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| *ToggleSky           | TS   | Switches the sky off. Also affects lighting, basically with switched off sky it's also "night".                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| ToggleTextureString  | TTS  | Unknown, I did not get any effect or output from this.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| ToggleWater          |      | Unknown, I did not get any effect or output from this.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| *ToggleWorld         | TW   | Stops the "world" and all objects from being rendered, just leaves the sky and water. Does not affect anything else, meaning collision, AI, etc. All remain active, just invisible                                                                                                                                                                                                                                                                                                                                                                     |

|                   |     |                                                                                                                                                                                                                        |
|-------------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ToggleWireframe   | TWF | Shows grid instead of full render                                                                                                                                                                                      |
| TogglePathGrid    | TPG | Toggle AI path grid display                                                                                                                                                                                            |
| *ToggleVanityMode | TVM | Switches to vanity mode, player can not switch again until toggle is off. May be useful for cutscenes.                                                                                                                 |
|                   | SA  | "Show Animation" Shows status and animation info for the selected actor: Spell readied, weapon drawn, attacked, animation group                                                                                        |
| ShowGroup         | SG  | Writes the selected Actors group actual ID + reference number to the console window. For the player this is PlayerSaveGame                                                                                             |
|                   | ST  | "Show target group" Writes the ID + reference number of the selected actors "target" group (combat opponent) to the console window.                                                                                    |
| ShowScenegraph    | SSG | Opens a new window that displays a hierarchical view of (I assume) the data structure of the currently rendered scene. Use TAB to get to the window. MW will remain unresponsive while the scene graph window is open. |

## Game Settings

The following long table is a list of game settings. Listed here are all settings that have a numerical value. Not listed are string entries – which are used to set many standard message texts, menu-texts, spell effect names, etc. But since they are fairly descriptive, they should be easy to figure out. Not so the numerical settings. Thanks to four forum members (maxpublic, Ldone, Wakim and Iudas), the meaning of many of the settings is now known and compiled into the list below. The list may still be a bit rough. I have not edited it thoroughly, but I am sure it will be interesting information for many modders.

In many cases where you see Base and Mult game settings, the formula they're involved in is a standard linear equation in the form  $y = mx + b$ , where  $m$  is the mult,  $b$  is the base, and  $x$  is some attribute, skill, or other value.

Note that the entry names begin with  $f$  or  $i$  (for floats and integers). The string entries begin with  $s$  (string). To my knowledge they can not be changed in-game, only in the TES-CS.

| Nr.   | Name                         | Value      | Description and Comments                                                                                                                                                                                                                                                                                                                  |
|-------|------------------------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| "465" | "fRepairMult"                | 1.0000     | Sets the general effectiveness of the repair skill of the character, via the armorer's hammer used                                                                                                                                                                                                                                        |
| "466" | "fRepairAmountMult"          | 3.0000     | Tells the game how many points of health are returned to the item when repaired<br>Determines cost for repairing items (Whether calculated from Max Item Health or Item Cost, I'm not sure)                                                                                                                                               |
| "467" | "fSpellValueMult"            | 10.0000    |                                                                                                                                                                                                                                                                                                                                           |
| "468" | "fSpellMakingValueMult"      | 7.0000     |                                                                                                                                                                                                                                                                                                                                           |
| "469" | "fEnchantmentValueMult"      | 1000.0000  | The setting for the price you pay at an enchanter to enchant an item. Linear.                                                                                                                                                                                                                                                             |
| "470" | "fTravelMult"                | 4000.0000  | Sets the cost of Silt Strider and boat travel (I think)<br>Multiplies cost of Travel – Unsure why the number is so high, but raising it raises the cost of Fast Travel                                                                                                                                                                    |
| "471" | "fTravelTimeMult"            | 16000.0000 | Tells the game how much time elapses during this sort of travel                                                                                                                                                                                                                                                                           |
| "472" | "fMagesGuildTravel"          | 10.0000    | Sets the cost of Guild Guide travel                                                                                                                                                                                                                                                                                                       |
| "947" | "fWortChanceValue"           | 15.0000    | Is compared to your alchemy skill to determine which of the effects of an ingredient you can see. -(Wakim, Iudas)                                                                                                                                                                                                                         |
| "949" | "fMinWalkSpeed"              | 100.0000   | This is the minimum walking speed of the PC, regardless of stats, skills or encumbrance                                                                                                                                                                                                                                                   |
| "950" | "fMaxWalkSpeed"              | 200.0000   | This is the maximum walking speed of the PC, regardless of stats, skills, or encumbrance<br>The actual walking speed of NPC's (and the PC) is set by checking various factors (Speed, Athletics, etc.) and assigning a value between fMinWalkSpeed and fMaxWalkSpeed based on that – The two settings dictate the spectrum of Walk Speeds |
| "951" | "fMinWalkSpeedCreature"      | 5.0000     | The same as for the PC, but if you badly encumber a creature it'll move veeerrry slowly. I've done this by accident.                                                                                                                                                                                                                      |
| "952" | "fMaxWalkSpeedCreature"      | 300.0000   | Same as above, they get faster, so they cover the speed spectrum more rapidly                                                                                                                                                                                                                                                             |
| "953" | "fEncumberedMoveEffect"      | 0.3000     | This sets how encumbrance affects walking and running speed, within the min/max limits set by other values.                                                                                                                                                                                                                               |
| "954" | "fBaseRunMultiplier"         | 1.7500     | Exactly as it says. Changing the value will increase/decrease base running speed.<br>Dictates how much faster Running is than the current Walk Speed                                                                                                                                                                                      |
| "955" | "fAthleticsRunBonus"         | 1.0000     | Sets how Athletics affects running speed.                                                                                                                                                                                                                                                                                                 |
| "956" | "fJumpAcrobaticsBase"        | 128.0000   | Sets the base jumping distance for the PC.                                                                                                                                                                                                                                                                                                |
| "957" | "fJumpAcroMultiplier"        | 4.0000     | Sets the multiplier for Acrobatics, which is why you can leap over tall buildings when your Acro is high enough.                                                                                                                                                                                                                          |
| "958" | "fJumpEncumbranceBase"       | 0.5000     | Effects how greatly jumping ability is effected by Encumbrance, but I'm unsure how                                                                                                                                                                                                                                                        |
| "959" | "fJumpEncumbranceMultiplier" | 1.0000     | Effects how greatly jumping ability is effected by Encumbrance, but I'm unsure how                                                                                                                                                                                                                                                        |
| "960" | "fJumpRunMultiplier"         | 1.0000     | UNSURE – Presumably effects Jump Distance while running (it doesn't seem to effect height, but I could be wrong)                                                                                                                                                                                                                          |
| "961" | "fJumpMoveBase"              | 0.5000     |                                                                                                                                                                                                                                                                                                                                           |
| "962" | "fJumpMoveMult"              | 0.5000     |                                                                                                                                                                                                                                                                                                                                           |
| "963" | "fSwimWalkBase"              | 0.5000     | Multiplies your walking speed to achieve the swimming speed at a 'walk'<br>Base swim speed while 'walking'                                                                                                                                                                                                                                |
| "964" | "fSwimRunBase"               | 0.5000     | Multiplies your running speed to achieve the swimming speed at a 'run'.                                                                                                                                                                                                                                                                   |

|        |                          |          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------|--------------------------|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| "965"  | "fSwimWalkAthleticsMult" | 0.0200   | Tells the game how Athletics affects 'walking' swimming speed. These low values keep you from flying through the water like you do on land when your Athletics is high.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| "966"  | "fSwimRunAthleticsMult"  | 0.1000   | Same as above.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| "967"  | "fSwimHeightScale"       | 0.9000   | Determines how close to the surface you have to be before the breathe indicator goes away                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| "968"  | "fHoldBreathTime"        | 20.0000  | The number of seconds your PC can hold her breath.<br>Base time that a character can remain underwater before incurring suffocation damage                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| "969"  | "fHoldBreathEndMult"     | 0.5000   | How Endurance affects the time you can hold your breath. I believe this is a flat-out multiplier to End, added as seconds to HoldBreathTime.<Doesn't seem to work.>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| "970"  | "fSuffocationDamage"     | 3.0000   | The amount of health damage you take each second you suffocate                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| "971"  | "fMinFlySpeed"           | 5.0000   | Exactly as it says – minimum flying speed.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| "972"  | "fMaxFlySpeed"           | 300.0000 | See above                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| "973"  | "fStromWindSpeed"        | 0.7000   | UNSURE - Determines altered walk speed during an ash or blight storm, but I'm unsure how – Might be a separate value from that entirely – might determine speed of storm particles /sprites(Dust, etc.) Interesting (possibly related) note, while treading water in an ash storm, I noticed I was moving slightly.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| "974"  | "fStromWalkMult"         | 0.2500   | Determines altered walk speed during an ash or blight storm, but I'm unsure how uses the getwindspeed to lower the PC movement speed during storms...                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| "975"  | "fFallDamageDistanceMin" | 400.0000 | The minimum distance you have to fall before you take damage.<br>(Presumably in units) In game units each unit - .056 inches                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| "976"  | "fFallDistanceBase"      | 0.0000   | This will increase/decrease the distance needed to fall before you take damage.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "977"  | "fFallDistanceMult"      | 0.0700   | Higher you are the more damage you take when you hit                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "978"  | "fFallAcroBase"          | 0.2500   | Acrobatics skill increases the distance you can fall before you take damage.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| "979"  | "fFallAcroMult"          | 0.0100   | Has to do w/ how the Acrobatics skill effects Fall Distance and Damage, but unsure how                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| "980"  | "iMaxActivateDist"       | 192      | Maximum distance for the player to be able to 'Activate' an object - approx 9 feet                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| "981"  | "iMaxInfoDist"           | 192      | Maximum distance for an Info message (object/NPC/creature name, etc.) to pop up in the Player's view                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "982"  | "fVanityDelay"           | 30.0000  | Seconds until VanityMode begins – the camera starts circling the player if there is no input via mouse or keyboard.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| "983"  | "fMaxHeadTrackDistance"  | 400.0000 | IIRC, this is the maximum distance an NPC or creature can be away from another NPC or creature and still trigger the 'head follow' routine you sometimes see. Put your PC in Balmora, let it go to Vanity View and you'll see your PC watch passing NPCs and 'follow' their movements for a certain amount of time.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| "984"  | "fInteriorHeadTrackMult" | 0.5000   | UNSURE – something to do w/ the modifier for this in Interiors – Do they track at half-distance in Interiors?                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| "985"  | "iHelmWeight"            | 5        | These values are used to set what weights are used to determine whether a piece of armor is light, medium, or heavy. Altering a value alters these categories for *all* armor of that type *everywhere* in the game. Rather nice, actually; I used it in redux to set weight categories for all of my armor types across the board.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| "986"  | "iPauldronWeight"        | 10       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "987"  | "iCuirassWeight"         | 30       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "988"  | "iGauntletWeight"        | 5        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "989"  | "iGreavesWeight"         | 15       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "990"  | "iBootsWeight"           | 20       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "991"  | "iShieldWeight"          | 15       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "992"  | "fLightMaxMod"           | 0.6000   | These values are used in conjunction with armor weights to set the weight classes (Light, Medium, Heavy).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| "993"  | "fMedMaxMod"             | 0.9000   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "994"  | "fUnarmoredBase1"        | 0.1000   | These two values dictate the range of AR for characters going Unarmored, based on the Unarmored skill – As they are, the settings produce a Maximum Unarmored AR of 65 (at 100 Unarmored skill), which you can see is some for of multiplication between the two settings – Reversing the values produces the same effect , so it seems the values are interchangeable (unless I missed something – could be wrong) - Changing one to 1.000 and the other to 0.0650 results in a max Unarmored AR of 650 – The game multiplies the numbers and then multiplies the resulting number by 1000 to obtain the actual in-game Max Unarmored AR – Doesn't seem to effect Min AR independently, only max - Progression of AR (from low skill to high skill) appears to be hard-coded.<br>It has also been found that the Unarmored skill doesn't work at all UNLESS atleast one item of armor is worn. (Forum Info / The other Felix) |
| "995"  | "fUnarmoredBase2"        | 0.0650   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "996"  | "iBaseArmorSkill"        | 30       | The Skill Level where in-game armors reach their base (i.e. 'In-Editor') AR value – Example: Glass Armor has a Base AR of 50 – At Light Armor skill level 30 (as indicated above), it will read as having AR 50 in-game – Before Skill Level 30, armors have diminished AR's from their Base Value, and after Skill Level 30, armors have higher AR's than their base until Skill Level reaches 100 – I haven't figured out the game's scheme for determining the mult value yet                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| "997"  | "fBlockStillBonus"       | 1.2500   | UNSURE – Presumably the amount that standing still increases the chance to block                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| "998"  | "fDamageStrengthBase"    | 0.5000   | Your STR adds to the damage you do with weapons. This determines how much damage is added.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| "999"  | "fDamageStrengthMult"    | 0.1000   | Effects amount that Strength effects damage dealt in combat (Unsure of how this value relates to in-game effect)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| "1000" | "fSwingBlockBase"        | 1.0000   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "1001" | "fSwingBlockMult"        | 1.0000   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "1002" | "fFatigueBase"           | 1.2500   | How much fatigue you lose while walking. However, this appears to let you jump very high without getting hurt (Bug? – Forum Info / DinkumThinkum).<br>1002 – 1022 All effect 'Fatigue' in-game, obviously – For separate actions,                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |

|        |                              |         |                                                                                                                                                                                                                                                                                                                            |
|--------|------------------------------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|        |                              |         | although not all of them actually have an effect in-game (I've never been able to get spells to reduce fatigue) – They seem pretty self-explanatory, but I haven't tested them thoroughly                                                                                                                                  |
| "1003" | "fFatigueMult"               | 0.5000  | Used to determine successful chance of casting if you are fatigued                                                                                                                                                                                                                                                         |
| "1004" | "fFatigueReturnBase"         | 2.5000  | How much fatigue you regain per second. This is why you don't actually fatigue while walking.                                                                                                                                                                                                                              |
| "1005" | "fFatigueReturnMult"         | 0.0200  | How much fatigue returns per second while walking                                                                                                                                                                                                                                                                          |
| "1006" | "fEndFatigueMult"            | 0.0400  |                                                                                                                                                                                                                                                                                                                            |
| "1007" | "fFatigueAttackBase"         | 2.0000  | How much fatigue you lose with every melee attack you make                                                                                                                                                                                                                                                                 |
| "1008" | "fFatigueAttackMult"         | 0.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1009" | "fWeaponFatigueMult"         | 0.2500  |                                                                                                                                                                                                                                                                                                                            |
| "1010" | "fFatigueBlockBase"          | 4.0000  | How much fatigue you lose blocking with a shield.                                                                                                                                                                                                                                                                          |
| "1011" | "fFatigueBlockMult"          | 0.0000  | This will increase the amount of fatigue lost when blocking with a shield.                                                                                                                                                                                                                                                 |
| "1012" | "fWeaponFatigueBlockMult"    | 1.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1013" | "fFatigueRunBase"            | 5.0000  | How much fatigue you lose running.                                                                                                                                                                                                                                                                                         |
| "1014" | "fFatigueRunMult"            | 2.0000  | This one appears to work with encumbrance, the more encumbered the more fatigue you lose/second                                                                                                                                                                                                                            |
| "1015" | "fFatigueJumpBase"           | 5.0000  | How much fatigue you lose jumping.                                                                                                                                                                                                                                                                                         |
| "1016" | "fFatigueJumpMult"           | 0.0000  | Modifier for fatigue loss                                                                                                                                                                                                                                                                                                  |
| "1017" | "fFatigueSwimWalkBase"       | 2.5000  | How much fatigue you lose swimming at a 'walk'                                                                                                                                                                                                                                                                             |
| "1018" | "fFatigueSwimRunBase"        | 7.0000  | How much fatigue you lose swimming at a 'run'.                                                                                                                                                                                                                                                                             |
| "1019" | "fFatigueSwimWalkMult"       | 0.0000  | Modifier for fatigue loss                                                                                                                                                                                                                                                                                                  |
| "1020" | "fFatigueSwimRunMult"        | 0.0000  | Modifier for fatigue loss                                                                                                                                                                                                                                                                                                  |
| "1021" | "fFatigueSneakBase"          | 1.5000  | The base level of fatigue loss while sneaking                                                                                                                                                                                                                                                                              |
| "1022" | "fFatigueSneakMult"          | 1.5000  | Multiplier to that base level                                                                                                                                                                                                                                                                                              |
| "1023" | "fMinHandToHandMult"         | 0.1000  |                                                                                                                                                                                                                                                                                                                            |
| "1024" | "fMaxHandToHandMult"         | 0.5000  |                                                                                                                                                                                                                                                                                                                            |
| "1025" | "fHandtoHandHealthPer"       | 0.1000  |                                                                                                                                                                                                                                                                                                                            |
| "1026" | "fCombatInvisoMult"          | 0.2000  | Reduce the chance to hit the PC when he is chameleoned or invisible                                                                                                                                                                                                                                                        |
| "1027" | "fCombatKODamageMult"        | 1.5000  |                                                                                                                                                                                                                                                                                                                            |
| "1028" | "fCombatCriticalStrikeMult"  | 4.0000  | This one appears to only work if you hit someone unawares while sneaking. I never got it to do anything else. 4x damage from a successful sneak attack works when chameleoned or invisible                                                                                                                                 |
| "1029" | "iBlockMinChance"            | 10      | Minimum chance of blocking with a shield                                                                                                                                                                                                                                                                                   |
| "1030" | "iBlockMaxChance"            | 50      | Maximum chance of blocking with a shield                                                                                                                                                                                                                                                                                   |
| "1031" | "fLevelUpHealthEndMult"      | 0.1000  | Multiplies current END to get hit points added at level-up.                                                                                                                                                                                                                                                                |
| "1032" | "fSoulGemMult"               | 3.0000  | A soul gem's monetary value is multiplied by this value to determine the soul capacity of a soul gem. Creatures with a soul value less than or equal to that capacity can "fit" in the gem.                                                                                                                                |
| "1033" | "fEffectCostMult"            | 0.5000  | The setting for all magicka costs for all spell effects. Changing this will change what all spells and enchantments cost. Everything. Linear change. Doubling this makes all spells cost twice as much magicka, all enchanted items cost twice as many charges.                                                            |
| "1034" | "fSpellPriceMult"            | 2.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1035" | "fFatigueSpellBase"          | 0.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1036" | "fFatigueSpellMult"          | 0.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1037" | "fFatigueSpellCostMult"      | 0.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1038" | "fPotionStrengthMult"        | 0.5000  |                                                                                                                                                                                                                                                                                                                            |
| "1039" | "fPotionT1MagMult"           | 1.5000  |                                                                                                                                                                                                                                                                                                                            |
| "1040" | "fPotionT1DurMult"           | 0.5000  |                                                                                                                                                                                                                                                                                                                            |
| "1041" | "fPotionMinUsefulDuration"   | 20.0000 |                                                                                                                                                                                                                                                                                                                            |
| "1042" | "fPotionT4BaseStrengthMult"  | 20.0000 |                                                                                                                                                                                                                                                                                                                            |
| "1043" | "fPotionT4EquipStrengthMult" | 12.0000 |                                                                                                                                                                                                                                                                                                                            |
| "1044" | "fIngredientMult"            | 1.0000  | Min # of an ingredient required to make a potion                                                                                                                                                                                                                                                                           |
| "1045" | "fMagicItemCostMult"         | 1.0000  | UNUSED                                                                                                                                                                                                                                                                                                                     |
| "1046" | "fMagicItemPriceMult"        | 1.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1047" | "fMagicItemOnceMult"         | 1.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1048" | "fMagicItemUsedMult"         | 1.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1049" | "fMagicItemStrikeMult"       | 1.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1050" | "fMagicItemConstantMult"     | 1.0000  |                                                                                                                                                                                                                                                                                                                            |
| "1051" | "fEnchantmentMult"           | 0.1000  | Setting for how much enchantment an item can hold based upon the value set in each individual item's property file. Linear, if an item in TESCS shows an enchantment value of 1200 (i.e. an exquisite ring) multiply it by fEnchantmentMult to get the actual enchantment you'll see in the make an enchanted item window. |
| "1052" | "fEnchantmentChanceMult"     | 3.0000  | These affect the PC's chance of making an enchantment                                                                                                                                                                                                                                                                      |
| "1053" | "fPCbaseMagickaMult"         | 1.0000  | This sets the spell point multiplier for the PC with respect to INT (e.g., 1 x INT with this setting)                                                                                                                                                                                                                      |
| "1054" | "fNPCbaseMagickaMult"        | 2.0000  | This does the same thing for NPCs.                                                                                                                                                                                                                                                                                         |
| "1055" | "fAutoSpellChance"           | 80.0000 |                                                                                                                                                                                                                                                                                                                            |
| "1056" | "fAutoPCSpellChance"         | 50.0000 |                                                                                                                                                                                                                                                                                                                            |
| "1057" | "iAutoSpellTimesCanCast"     | 3       |                                                                                                                                                                                                                                                                                                                            |
| "1058" | "iAutoSpellAttSkillMin"      | 70      |                                                                                                                                                                                                                                                                                                                            |
| "1059" | "iAutoSpellAlterationMax"    | 5       |                                                                                                                                                                                                                                                                                                                            |

|        |                                    |          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------|------------------------------------|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| "1060" | "iAutoSpellConjurationMax"         | 2        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1061" | "iAutoSpellDestructionMax"         | 5        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1062" | "iAutoSpellIllusionMax"            | 5        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1063" | "iAutoSpellMysticismMax"           | 5        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1064" | "iAutoSpellRestorationMax"         | 5        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1065" | "iAutoPCSpellMax"                  | 100      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1066" | "iAutoRepFacMod"                   | 2        | A positive modification to relations you get with people who belong to the same faction                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| "1067" | "iAutoRepLevMod"                   | 0        | You can apparently add rep points with each level-up. I've never tried it.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "1068" | "iMagicItemChargeOnce"             | 1        | 1068-1071 effect the amount of charges auto-calculated on magic items based on their function – This value is the number of uses that the game will account for when calculating the max charges of a magic item (works universally, across the board with all –ingame items –<br>1068 is the setting for the number of charges an automatically calculated cast once effect enchanted item will have. Formula is BaseSpellEffectCost x iMagicItemChargeOnce. Linear. This way an item with a cast once effect will have exactly the number of charges needed to cast the effect upon it<br>1069 for const effect items<br>1070 is the setting for the multiplier for charges for automatically calculated cast when used effect enchanted items. See above for explanation.<br>1071 for "Cast on Strike" items<br>(charges are calculated to account for X 'uses' with this value as is) |
| "1069" | "iMagicItemChargeConst"            | 10       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1070" | "iMagicItemChargeUse"              | 5        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1071" | "iMagicItemChargeStrike"           | 10       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1072" | "iMonthsToRespawn"                 | 4        | The time to respawn things like the Fighters Guild/Mages Guild chests, etc. How many months before a picked plant respawns ingredients. Chests in guilds respawn contents the same as any other chests.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| "1073" | "fCorpseClearDelay"                | 72.0000  | How many hours it takes before a non-persistent corpse disappears. Also controls time before certain temporary data is cleared if the object has not been in an active cell in that time (e.g. Talked to PC flag).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| "1074" | "fCorpseRespawnDelay"              | 72.0000  | How many hours it takes before a respawnable creature actually respawns (note that this doesn't seem to work properly).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| "1075" | "fBarterGoldResetDelay"            | 24.0000  | How many hours it takes before a trader resets it barter gold to its default value.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| "1076" | "fEncumbranceStrMult"              | 5.0000   | A straight multiplier to STR to see how much a PC/NPC/creature can carry.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| "1077" | "fPickLockMult"                    | -1.0000  | Dictates amount that Lock pick difficulty raises according to Lock Level – Lower the value here, the harder it gets – Positive values make locks get easier with higher lock Levels                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| "1078" | "fTrapCostMult"                    | 0.0000   | Dictates difficulty of traps based on the spell cost of the spell assigned as trap – Lower the value here, the harder it gets to disarm (again, based on spell cost of assigned 'trap')<br>The values is multiplied by the spell cost of a trap and then added to your chance of disarming it. Since it's set to zero, the trap spell's cost is not incorporated into the chance. So basically it's also unused.                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| "1079" | "fMessageTimePerChar"              | 0.1000   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1080" | "fMagicItemRechargePerSecond"      | 0.0500   | This is the setting for the amount of charges restored to a charged magic item per second of game play. Linear. 0.05 x 20 seconds = 1 charge restored.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| "1081" | "i1stPersonSneakDelta"             | 10       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1082" | "iBarterSuccessDisposition"        | 1        | If you barter with a merchant successfully, your disposition with that merchant increases by one and falls by 1 if you fail a barter attempt.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| "1083" | "iBarterFailDisposition"           | -1       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1084" | "iLevelupTotal"                    | 10       | How many skill points you need before you level up.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| "1085" | "iLevelupMajorMult"                | 1        | How much each major skill is worth in points. E.g., if you set this to 2 then each point earned in a major skill counts as 2 skill points for leveling up.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| "1086" | "iLevelupMinorMult"                | 1        | Same as above, but for minor skills.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| "1087" | "iLevelupMajorMultAttribute"       | 1        | I *think* - not sure if I remember this correctly - but I think this works like the above, but for skills governed by your two primary attributes. So if your primary attributes are STR and AGI, and you set 1087 to 2, then any major skill governed by one of these attributes which goes up by a point counts as 2 points for purposes of leveling.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| "1088" | "iLevelupMinorMultAttribute"       | 1        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1089" | "iLevelupMiscMultAttribute"        | 1        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1090" | "iLevelupSpecialization"           | 1        | The game keeps track of how many skill points you've gained since the last level up. If you gained 8 skill points in skills governed by AGI, then when you get to distribute attribute points whatever number is in place for iLevelUp08Mult will be used for AGI. So if this value is 4, you'll see a x 4 next to AGI when you level up (you'll get 4 points in AGI if you pick this during the leveling process).                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| "1091" | "iLevelUp01Mult"                   | 2        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1092" | "iLevelUp02Mult"                   | 2        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1093" | "iLevelUp03Mult"                   | 2        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1094" | "iLevelUp04Mult"                   | 2        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1095" | "iLevelUp05Mult"                   | 3        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1096" | "iLevelUp06Mult"                   | 3        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1097" | "iLevelUp07Mult"                   | 3        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1098" | "iLevelUp08Mult"                   | 4        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1099" | "iLevelUp09Mult"                   | 4        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1100" | "iLevelUp10Mult"                   | 5        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1101" | "iSoulAmountForConstantEffect"     | 400      | This is the setting for the minimum soul value to toggle the constant effect button in the enchantment creation window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| "1102" | "fConstantEffectMult"              | 15.0000  | UNUSED                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| "1103" | "fEnchantmentConstantDurationMult" | 100.0000 | This setting is the multiplier for constant effect cast cost as compared to a 0 duration spell. so restore health 2-2 for 0 secs, which costs 0.50 to cast as a spell, costs 0.5 x 100 = 50 as a constant effect.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| "1104" | "fEnchantmentConstantChanceMult"   | 0.5000   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| "1105" | "fWeaponDamageMult"                | 0.1000   | weapon damage during combat. depreciation as it were.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

|        |                                |           |                                                                                                                                                                                                                                                                                                    |
|--------|--------------------------------|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| "1106" | "fSeriousWoundMult"            | 0.0000    | UNUSED                                                                                                                                                                                                                                                                                             |
| "1107" | "fKnockDownMult"               | 0.5000    | This sets the chance for a knock-down factored on how much damage you do in a single blow.                                                                                                                                                                                                         |
| "1108" | "iKnockDownOddsBase"           | 50        | Sets the base odds for a knockdown when the condition for it is met                                                                                                                                                                                                                                |
| "1109" | "iKnockDownOddsMult"           | 50        |                                                                                                                                                                                                                                                                                                    |
| "1110" | "fCombatArmorMinMult"          | 0.2500    |                                                                                                                                                                                                                                                                                                    |
| "1111" | "fHandToHandReach"             | 1.0000    | Sets the reach of HTH weapons. Values of less than 1.0 have no meaning.                                                                                                                                                                                                                            |
| "1112" | "fVoiceIdleOdds"               | 10.0000   | Controls likelihood of an NPC 'speaking' a voice clip when Idle (Unsure of specifics)                                                                                                                                                                                                              |
| "1113" | "iVoiceAttackOdds"             | 10        | Controls likelihood of an NPC 'speaking' a voice clip when attacking (Unsure of specifics)                                                                                                                                                                                                         |
| "1114" | "iVoiceHitOdds"                | 30        | Controls likelihood of an NPC 'speaking' a voice clip when being hit (Unsure of specifics)                                                                                                                                                                                                         |
| "1115" | "fProjectileMinSpeed"          | 400.0000  | Sets the minimum speed of projectile weapons                                                                                                                                                                                                                                                       |
| "1116" | "fProjectileMaxSpeed"          | 3000.0000 | Dictates maximum speed of projectiles from bows and crossbows                                                                                                                                                                                                                                      |
| "1117" | "fThrownWeaponMinSpeed"        | 300.0000  | Sets the minimum speed of thrown weapons                                                                                                                                                                                                                                                           |
| "1118" | "fThrownWeaponMaxSpeed"        | 1000.0000 | Dictates Max speed of thrown weapons                                                                                                                                                                                                                                                               |
| "1119" | "fTargetSpellMaxSpeed"         | 1000.0000 | Sets the speed of spells. Double this and your spells will *zip* across the screen! – Min speed is apparently hard-coded                                                                                                                                                                           |
| "1120" | "fProjectileThrownStoreChance" | 25.0000   | The odds of getting arrows back when you loot a corpse. Thrown weapons also                                                                                                                                                                                                                        |
| "1121" | "iPickMinChance"               | 5         | Minimum pickpocketing chance (forum info / Iudas)                                                                                                                                                                                                                                                  |
| "1122" | "iPickMaxChance"               | 75        | Maximum pickpocketing chance (forum info / Iudas)                                                                                                                                                                                                                                                  |
| "1123" | "fDispRaceMod"                 | 5.0000    | You have better relations with your own race than with others.                                                                                                                                                                                                                                     |
| "1124" | "fDispPersonalityMult"         | 0.5000    | These determine how personality affect NPC disposition                                                                                                                                                                                                                                             |
| "1125" | "fDispPersonalityBase"         | 50.0000   |                                                                                                                                                                                                                                                                                                    |
| "1126" | "fDispFactionMod"              | 3.0000    | These determine how your rank in a faction will alter your relations with people that belong to that faction. This is why when you reach high ranks in a faction everyone in that faction suddenly becomes very friendly.                                                                          |
| "1127" | "fDispFactionRankBase"         | 1.0000    |                                                                                                                                                                                                                                                                                                    |
| "1128" | "fDispFactionRankMult"         | 0.5000    |                                                                                                                                                                                                                                                                                                    |
| "1129" | "fDispCrimeMod"                | 0.0000    | This is multiplied by the player's crime level (bounty) to determine how that information affects an NPC's disposition towards the player.                                                                                                                                                         |
| "1130" | "fDispDiseaseMod"              | -10.0000  | How much disposition is lowered when you're suffering from a disease.                                                                                                                                                                                                                              |
| "1131" | "iDispAttackMod"               | -50       | Not completely sure – NPC disposition modifier if PC attacks said NPC                                                                                                                                                                                                                              |
| "1132" | "fDispWeaponDrawn"             | -5.0000   | How much disposition is lowered when you have a weapon drawn.                                                                                                                                                                                                                                      |
| "1133" | "fDispBargainSuccessMod"       | 1.0000    | I don't remember if these work the same as the previous barter disposition values, or if these are multipliers. These effect the long term disposition of the merchant                                                                                                                             |
| "1134" | "fDispBargainFailMod"          | -1.0000   |                                                                                                                                                                                                                                                                                                    |
| "1135" | "fDispPickPocketMod"           | -25.0000  | NPC disposition modifier for catching the PC attempting to pickpocket them                                                                                                                                                                                                                         |
| "1136" | "iDaysinPrisonMod"             | 100       | Determines prison time based on your crime level.                                                                                                                                                                                                                                                  |
| "1137" | "fDispAttacking"               | -10.0000  | Unsure – I believe it's an NPC Disposition modifier if the PC is attacking something other than the NPC, ie it effects non-combatants disposition.                                                                                                                                                 |
| "1138" | "fDispStealing"                | -0.5000   | Unsure - I believe it's an NPC Disposition modifier if the PC is stealing from someone other than the NPC it effects                                                                                                                                                                               |
| "1139" | "iDispTresspass"               | -20       | NPC Disposition modifier for catching the PC 'trespassing' – Not sure what that exactly means in-game                                                                                                                                                                                              |
| "1140" | "iDispKilling"                 | -50       | Unsure NPC Disposition modifier for witnessing the PC kill an innocent NPC (I think...)                                                                                                                                                                                                            |
| "1141" | "iTrainingMod"                 | 10        | Determines training costs. The higher the value, the more training costs. – unsure of method of calculation                                                                                                                                                                                        |
| "1142" | "iAlchemyMod"                  | 2         | Controls the value of player-made potions. Set to 0 and player made potions have 0 value, set to 1 and player made potions have about 1/4 to 1/2 of what they have if you leave it at the default 2 (forum info / BeanCounter, Iudas).                                                             |
| "1143" | "fBargainOfferBase"            | 50.0000   | This is multiplied by the item's value to determine what the merchant will offer when selling. Base value is also modified by PC level. Base amount that merchants will buy items from you for, in percentage points – Believe it goes both ways, but I'm unsure how it would work the 'other way' |
| "1144" | "fBargainOfferMulti"           | -4.0000   | Effects how much the merchant lowers his offers during a bargaining session                                                                                                                                                                                                                        |
| "1145" | "fDispositionMod"              | 1.0000    |                                                                                                                                                                                                                                                                                                    |
| "1146" | "fPersonalityMod"              | 5.0000    |                                                                                                                                                                                                                                                                                                    |
| "1147" | "fLuckMod"                     | 10.0000   | IIRC, this is multiplied by your Luck as a percentage to get a base increase to all skills.                                                                                                                                                                                                        |
| "1148" | "fReputationMod"               | 1.0000    |                                                                                                                                                                                                                                                                                                    |
| "1149" | "fLevelMod"                    | 5.0000    |                                                                                                                                                                                                                                                                                                    |
| "1150" | "fBribe10Mod"                  | 35.0000   | Dictates amount that NPC Disposition will raise on a successful 10 Gold Bribe – Don't believe it's in straight disposition points – Could be percentages - (Other factors like race, sex, opposing faction etc. reduce this amount significantly)                                                  |
| "1151" | "fBribe100Mod"                 | 75.0000   | Dictates amount that NPC Disposition will raise on a successful 100 Gold Bribe. See above.                                                                                                                                                                                                         |
| "1152" | "fBribe1000Mod"                | 150.0000  | Dictates amount that NPC Disposition will raise on a successful 1000 Gold Bribe. See above.                                                                                                                                                                                                        |
| "1153" | "fPerDieRollMult"              | 0.3000    |                                                                                                                                                                                                                                                                                                    |
| "1154" | "fPerTempMult"                 | 1.0000    | is used in just about every disposition modifying calculation.                                                                                                                                                                                                                                     |
| "1155" | "iPerMinChance"                | 5         |                                                                                                                                                                                                                                                                                                    |
| "1156" | "iPerMinChange"                | 10        |                                                                                                                                                                                                                                                                                                    |
| "1157" | "fSpecialSkillBonus"           | 0.8000    | These all determine how fast you gain skill points in each skill. The lower the value, the faster you'll gain skill points. The values are multiplied by whatever rate                                                                                                                             |
| "1158" | "fMajorSkillBonus"             | 0.7500    |                                                                                                                                                                                                                                                                                                    |

|        |                             |           |                                                                                                                                                                                                                                                                                                                                                                                     |
|--------|-----------------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| "1159" | "fMinorSkillBonus"          | 1.0000    | you set for each individual skill.                                                                                                                                                                                                                                                                                                                                                  |
| "1160" | "fMiscSkillBonus"           | 1.2500    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1161" | "iAlarmKilling"             | 90        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1162" | "iAlarmAttack"              | 50        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1163" | "iAlarmStealing"            | 1         |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1164" | "iAlarmPickPocket"          | 20        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1165" | "iAlarmTresspass"           | 5         |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1166" | "fAlarmRadius"              | 2000.0000 | When an NPC raises the alarm, this is the base radius for response by other affiliated NPCs.                                                                                                                                                                                                                                                                                        |
| "1167" | "iCrimeKilling"             | 1000      | These set the gold value for crimes. I believe that fCrimeStealing is multiplied by the price of the item stolen.                                                                                                                                                                                                                                                                   |
| "1168" | "iCrimeAttack"              | 40        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1169" | "fCrimeStealing"            | 1.0000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1170" | "iCrimePickPocket"          | 25        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1171" | "iCrimeTresspass"           | 5         |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1172" | "iCrimeThreshold"           | 1000      | When NPC's start to react negatively to the PC                                                                                                                                                                                                                                                                                                                                      |
| "1173" | "iCrimeThresholdMultiplier" | 10        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1174" | "fCrimeGoldDiscountMult"    | 0.5000    | Thieves guild discount when you have a price on your head.                                                                                                                                                                                                                                                                                                                          |
| "1175" | "fCrimeGoldTurnInMult"      | 0.9000    | Discount on the fine if you turn yourself in.                                                                                                                                                                                                                                                                                                                                       |
| "1176" | "iFightAttack"              | 100       |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1177" | "iFightAttacking"           | 50        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1178" | "iFightDistanceBase"        | 20        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1179" | "fFightDistanceMultiplier"  | 0.0050    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1180" | "iFightAlarmMult"           | 1         |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1181" | "fFightDispMult"            | 0.2000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1182" | "fFightStealing"            | 50.0000   |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1183" | "iFightPickpocket"          | 25        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1184" | "iFightTresspass"           | 25        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1185" | "iFightKilling"             | 50        |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1186" | "iFlee"                     | 0         | UNUSED                                                                                                                                                                                                                                                                                                                                                                              |
| "1187" | "iGreetDistanceMultiplier"  | 6         | Used for those annoying voice greetings NPCs use when you get too close Specifically (if the Construction Set help is to be believed) this is multiplied by their hello rating to get the distance before they talk.                                                                                                                                                                |
| "1188" | "iGreetDuration"            | 4         |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1189" | "fGreetDistanceReset"       | 512.0000  | How far away from an NPC you have to get before they check for a voice greeting again.                                                                                                                                                                                                                                                                                              |
| "1190" | "fIdleChanceMultiplier"     | 0.7500    | Probability multiplier that an NPC will mumble something while standing idly near the PC                                                                                                                                                                                                                                                                                            |
| "1191" | "fSneakUseDist"             | 500.0000  | Helps determine if you can sneak                                                                                                                                                                                                                                                                                                                                                    |
| "1192" | "fSneakUseDelay"            | 1.0000    | Helps determine how long before the Sneak Icon come on                                                                                                                                                                                                                                                                                                                              |
| "1193" | "fSneakDistanceBase"        | 0.5000    | see above                                                                                                                                                                                                                                                                                                                                                                           |
| "1194" | "fSneakDistanceMultiplier"  | 0.0020    | see above                                                                                                                                                                                                                                                                                                                                                                           |
| "1195" | "fSneakSpeedMultiplier"     | 0.7500    | Multiplied by base walking speed to see how fast you move while sneaking.                                                                                                                                                                                                                                                                                                           |
| "1196" | "fSneakViewMult"            | 1.5000    | Makes it more difficult to sneak when in view of an NPC.                                                                                                                                                                                                                                                                                                                            |
| "1197" | "fSneakNoViewMult"          | 0.5000    | Makes it easier to sneak when you aren't in view.                                                                                                                                                                                                                                                                                                                                   |
| "1198" | "fSneakSkillMult"           | 1.0000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1199" | "fSneakBootMult"            | -1.0000   | Multiplied by the boot value (weight?) to determine the reduction to Sneak skill.                                                                                                                                                                                                                                                                                                   |
| "1200" | "fCombatDistance"           | 128.0000  | Combined with weapon reach, determines the effective distance that hits can be obtained                                                                                                                                                                                                                                                                                             |
| "1201" | "fCombatAngleXY"            | 60.0000   |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1202" | "fCombatAngleZ"             | 60.0000   |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1203" | "fCombatForceSideAngle"     | 30.0000   |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1204" | "fCombatTorsoSideAngle"     | 45.0000   |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1205" | "fCombatTorsoStartPercent"  | 0.3000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1206" | "fCombatTorsoStopPercent"   | 0.8000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1207" | "fCombatBlockLeftAngle"     | -90.0000  | Shields are worn on the left and partially block attacks from 90 degrees left to 30 degrees right of the PC's facing.                                                                                                                                                                                                                                                               |
| "1208" | "fCombatBlockRightAngle"    | 30.0000   |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1209" | "fCombatDelayCreature"      | 0.1000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1210" | "fCombatDelayNPC"           | 0.1000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1212" | "fAIMeleeWeaponMult"        | 2.0000    | Used in the determination of how far away an NPC will flee if they flee combat and the PC has a melee weapon in hand                                                                                                                                                                                                                                                                |
| "1213" | "fAIRangeMeleeWeaponMult"   | 5.0000    | as above but the PC has a crossbow or Bow in hand                                                                                                                                                                                                                                                                                                                                   |
| "1214" | "fAIMagicSpellMult"         | 3.0000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1215" | "fAIRangeMagicSpellMult"    | 5.0000    | As above but the PC has a spell readied                                                                                                                                                                                                                                                                                                                                             |
| "1216" | "fAIMeleeArmorMult"         | 1.0000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1217" | "fAIMeleeSummWeaponMult"    | 1.0000    |                                                                                                                                                                                                                                                                                                                                                                                     |
| "1218" | "fAIFleeHealthMult"         | 7.0000    | Alters the opponents flee rating when health declines                                                                                                                                                                                                                                                                                                                               |
| "1219" | "fAIFleeFleeMult"           | 0.3000    | Used to alter base flee ratings.                                                                                                                                                                                                                                                                                                                                                    |
| "1220" | "fPickPocketMod"            | 0.3000    | Multiplier to item's weight vs player's security. As far as I've tested it, chances are as follows:<br>Enter/Leave inventory:<br><i>Player.Sneak</i> > <i>d100</i> + <i>Victim.Sneak</i><br>Take items:<br><i>Player.Security</i> > <i>d100</i> + <i>fPickPocketMod</i> * <i>item Weight</i><br>In the moment you successfully take an item, the inventory is reloaded (this is why |

|        |                                   |           |                                                                                                                                                                                                                                                                                                                                                                   |
|--------|-----------------------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|        |                                   |           | sometimes items vanish or appear) and you have to make another Enter/Leave inventory check, so effectively both checks are made when taking items. When the victim can't detect you (sneaking, invisibility or chameleon) you always succeed in entering the inventory, though only chameleon will also help you in taking items and leaving the inventory. (JOG) |
| "1221" | "fSleepRandMod"                   | 0.2500    | Affects the chance of a mob waking the PC up while asleep in the wilderness.                                                                                                                                                                                                                                                                                      |
| "1222" | "fSleepRestMod"                   | 0.3000    | Unused (Thanks to Damar Stiehl for these two)                                                                                                                                                                                                                                                                                                                     |
| "1223" | "iNumberCreatures"                | 1         |                                                                                                                                                                                                                                                                                                                                                                   |
| "1224" | "fAudioDefaultMinDistance"        | 5.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
| "1225" | "fAudioDefaultMaxDistance"        | 40.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
| "1226" | "fAudioVoiceDefaultMinDistance"   | 10.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
| "1227" | "fAudioVoiceDefaultMaxDistance"   | 60.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
| "1228" | "fAudioMinDistanceMult"           | 20.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
| "1229" | "fAudioMaxDistanceMult"           | 50.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
| "1230" | "fNPCHealthBarTime"               | 3.0000    | Controls delay before the Opponents health bar disappears                                                                                                                                                                                                                                                                                                         |
| "1231" | "fNPCHealthBarFade"               | 0.5000    | Controls how many seconds the bar "fades" for (rather than abruptly vanishing)                                                                                                                                                                                                                                                                                    |
| "1232" | "fDifficultyMult"                 | 5.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
| "1399" | "fMagicDetectRefreshRate"         | 0.0167    |                                                                                                                                                                                                                                                                                                                                                                   |
| "1400" | "fMagicStartIconBlink"            | 3.0000    | The number of seconds a spell icon will fade before the spell runs out, on the lower right-hand corner of the screen.                                                                                                                                                                                                                                             |
| "1401" | "fMagicCreatureCastDelay"         | 1.5000    |                                                                                                                                                                                                                                                                                                                                                                   |
| "1431" | "fDiseaseXferChance"              | 2.5000    | The chance of catching a disease if hit by a creature, or looting a diseased creature's corpse.                                                                                                                                                                                                                                                                   |
| "1432" | "fElementalShieldMult"            | 0.1000    |                                                                                                                                                                                                                                                                                                                                                                   |
| "1435" | "fMagicSunBlockedMult"            | 0.5000    | Vampire weakness                                                                                                                                                                                                                                                                                                                                                  |
|        | "fWereWolfRunMult"                | 1.3000    | Werewolf run speed multiplier.                                                                                                                                                                                                                                                                                                                                    |
|        | "fWereWolfSilverWeaponDamageMult" | 2.0000    | The damage multiplier for silver weapon damage against all werewolves.                                                                                                                                                                                                                                                                                            |
|        | "iWereWolfBounty"                 | 1000      | These are the skills and attributes for Werewolf form.                                                                                                                                                                                                                                                                                                            |
|        | "fWereWolfStrength"               | 150.0000  |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfAgility"                | 150.0000  |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfEndurance"              | 150.0000  |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfSpeed"                  | 90.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfHandtoHand"             | 100.0000  |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfUnarmored"              | 100.0000  |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfAthletics"              | 50.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfAcrobatics"             | 80.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfIntelligence"           | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfWillPower"              | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfPersonality"            | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfLuck"                   | 25.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfBlock"                  | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfArmorer"                | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfMediumArmor"            | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfHeavyArmor"             | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfBluntWeapon"            | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfLongBlade"              | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfAxe"                    | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfSpear"                  | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfDestruction"            | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfAlteration"             | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfIllusion"               | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfConjuration"            | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfMysticism"              | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfRestoration"            | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfEnchant"                | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfAlchemy"                | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfSecurity"               | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfSneak"                  | 95.0000   |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfLightArmor"             | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfShortBlade"             | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfMarksman"               | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfSpeechcraft"            | 0.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "iWereWolfLevelToAttack"          | 20        | Determines the attack range of a Werewolf.                                                                                                                                                                                                                                                                                                                        |
|        | "iWereWolfFightMod"               | 100       |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "iWereWolfFleeMod"                | 100       |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfHealth"                 | 2.0000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfFatigue"                | 400.0000  |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fWereWolfMagica"                 | 100.0000  | Determines how far away someone will flee.                                                                                                                                                                                                                                                                                                                        |
|        | "fCombatDistaceWereWolfMod"       | 0.3000    |                                                                                                                                                                                                                                                                                                                                                                   |
|        | "fFleeDistance"                   | 3000.0000 |                                                                                                                                                                                                                                                                                                                                                                   |

# Index

"

"Equal to", 32  
"Greater than or equal to", 32  
"Greater than", 32  
"Smaller than or equal to", 32  
"Smaller than", 32  
"Unequal to", 32

[

[no fix], 35

^

^Cell, 105  
^Class, 106  
^Faction, 106  
^Gamehour, 106  
^Global, 106  
^Name, 106  
^NextPCRank, 105  
^PCClass, 105  
^PCName, 105  
^PCRace, 105  
^PCRank, 105  
^Race, 106  
^Rank, 106

## A

Activate, 66  
AddItem, 36  
Addition, 30  
AddSoulGem, 132  
AddSpell, 133  
AddToLevCreature, 156  
AddToLevItem, 156  
AddTopic, 106  
AiActivate, 79  
AIEscort, 81  
AIEscortCell, 81  
AiFollow, 80  
AiFollowCell, 80  
AiTravel, 75  
AiWander, 77  
AllowWereWolfForceGreeting, 109, 209  
AND, 34

## B

BecomeWerewolf, 95  
Bloodmoon Script Functions, 209  
Boolean operators, 34

## C

Cast, 134  
CellChanged, 62  
**CellUpdate**, 48  
CenterOnCell, 216  
CenterOnExterior, 216  
ChangeWeather, 146

character creation, 150  
Choice, 109  
Cleaning, 163  
ClearForceJump, 85  
ClearForceMoveJump, 85  
ClearForceRun, 85  
ClearForceSneak, 83  
ClearInfoActor, 110  
coc, 216  
coe, 216  
**Commands**, 15  
Comments, 26  
companion, 88  
Console, 204  
console commands, 216  
cosine, 189  
CreateMaps, 216  
CrimeGoldDiscount, 128  
CrimeGoldTurnIn, 128  
ctrl-c, 14  
ctrl-v, 14  
ctrl-x, 14

## D

Day, 144  
DaysPassed, 144  
Disable, 71  
DisableLevitation, 130  
DisablePlayerControls, 149  
DisablePlayerFighting, 149  
DisablePlayerJumping, 149  
DisablePlayerLooking, 149  
DisablePlayerMagic, 149  
DisablePlayerViewSwitch, 149  
DisableTeleporting, 129  
DisableVanityMode, 149  
Division, 30  
DontSaveObject, 74  
Drop, 38  
DropSoulgem, 132

## E

Else, 31  
Elseif, 31  
Enable, 71  
EnableBirthMenu, 150  
EnableClassMenu, 150  
EnableInventoryMenu, 150  
EnableLevelUpMenu, 149  
EnableLevitation, 130  
EnableMagicMenu, 150  
EnableMapMenu, 150  
EnableNameMenu, 150  
EnablePlayerControls, 150  
EnablePlayerFighting, 150  
EnablePlayerJumping, 150  
EnablePlayerLooking, 150  
EnablePlayerMagic, 150  
EnablePlayerViewSwitch, 150  
EnableRaceMenu, 150  
EnableRest, 150  
EnableStatsMenu, 150  
EnableTeleporting, 129

EnableVanityMode, 150  
Endif, 31  
EndWhile, 33  
Equip, 40  
Error messages, 205  
ExplodeSpell, 138  
**EXPRESSION**, 205

## F

Face, 77  
FadeIn, 154  
FadeOut, 154  
FadeTo, 154  
Fall, 88  
FixMe, 216  
Float, 27  
ForceGreeting, 107  
ForceJump, 85  
ForceMoveJump, 85  
ForceRun, 85  
ForceSneak, 83  
Friend Hit (dialogue), 112  
**Functions**, 15

## G

game settings, 219  
GameHour, 144  
*Get/Mod/SetAcrobatics*, 115  
*Get/Mod/SetAgility*, 114  
*Get/Mod/SetAlarm*, 123  
*Get/Mod/SetAlchemy*, 115  
*Get/Mod/SetAlteration*, 115  
*Get/Mod/SetArmorBonus*, 139  
*Get/Mod/SetArmorer*, 115  
*Get/Mod/SetAthletics*, 115  
*Get/Mod/SetAttackBonus*, 139  
*Get/Mod/SetAxe*, 115  
*Get/Mod/SetBlindness*, 139  
*Get/Mod/SetBlock*, 115  
*Get/Mod/SetBluntWeapon*, 115  
*Get/Mod/SetCastPenalty*, 139  
*Get/Mod/SetConjuration*, 115  
*Get/Mod/SetDefendBonus*, 139  
*Get/Mod/SetDestruction*, 115  
*Get/Mod/SetDisposition*, 93  
*Get/Mod/SetEnchant*, 115  
*Get/Mod/SetEndurance*, 114  
*Get/Mod/SetFatigue*, 114  
*Get/Mod/SetFight*, 122  
*Get/Mod/SetFlee*, 122  
*Get/Mod/SetFlying*, 139  
*Get/Mod/SetHandToHand*, 115  
*Get/Mod/SetHealth*, 114  
*Get/Mod/SetHeavyArmor*, 115  
*Get/Mod/SetHello*, 112  
*Get/Mod/SetIllusion*, 115  
*Get/Mod/SetIntelligence*, 114  
*Get/Mod/SetInvisible*, 139  
*Get/Mod/SetInvisible*, 139  
*Get/Mod/SetLevel*, 115  
*Get/Mod/SetLightArmor*, 115  
*Get/Mod/SetLongBlade*, 115  
*Get/Mod/SetLuck*, 114  
*Get/Mod/SetMagicka*, 114  
*Get/Mod/SetMarksman*, 115  
*Get/Mod/SetMediumArmor*, 115

*Get/Mod/SetMercantile*, 115  
*Get/Mod/SetMysticism*, 115  
*Get/Mod/SetPCCrimeLevel*, 126  
*Get/Mod/SetPersonality*, 114  
*Get/Mod/SetReputation*, 93  
*Get/Mod/SetResistBlight*, 138  
*Get/Mod/SetResistCorpus*, 138  
*Get/Mod/SetResistDisease*, 138  
*Get/Mod/SetResistFire*, 138  
*Get/Mod/SetResistFrost*, 138  
*Get/Mod/SetResistMagicka*, 138  
*Get/Mod/SetResistNormalWeapons*, 138  
*Get/Mod/SetResistParalysis*, 138  
*Get/Mod/SetResistPoison*, 138  
*Get/Mod/SetResistShock*, 138  
*Get/Mod/SetRestoration*, 115  
*Get/Mod/SetSecurity*, 115  
*Get/Mod/SetShortBlade*, 115  
*Get/Mod/SetSilence*, 139  
*Get/Mod/SetSneak*, 115  
*Get/Mod/SetSpear*, 115  
*Get/Mod/SetSpeechcraft*, 115  
*Get/Mod/SetSpeed*, 114  
*Get/Mod/SetStrength*, 114  
*Get/Mod/SetSuperJump*, 139  
*Get/Mod/SetSwimSpeed*, 139  
*Get/Mod/SetUnarmored*, 115  
*Get/Mod/SetWaterBreathing*, 138  
*Get/Mod/SetWaterWalking*, 139  
*Get/Mod/SetWillpower*, 114  
GetAIPackageDone, 75  
GetAngle, 60  
GetArmorType, 43  
GetAttacked, 121  
GetBlightDisease, 137  
GetButtonPressed, 105  
GetCollidingActor, 65  
GetCollidingPC, 65  
GetCommonDisease, 137  
GetCurrentAIPackage, 81  
GetCurrentTime, 143  
GetCurrentWeather, 147  
GetDeadCount, 124  
GetDetected, 62  
GetDisabled, 71  
GetDistance, 59  
GetEffect, 136  
GetFactionReaction, 216  
GetForceJump, 85  
GetForceMoveJump, 85  
GetForceRun, 85  
GetForceSneak, 83  
GetHealthGetRatio, 114  
GetInterior, 58  
GetItemCount, 43  
GetJournalIndex, 110  
GetLineOfSight, 61  
GetLocked, 69  
GetLOS, 60  
GetMasserPhase, 144  
GetPCCell, 58  
GetPCCrimeLevel, 127  
GetPCFacRep, 91  
GetPCInJail, 127  
GetPCJumping, 86  
GetPCRank, 90  
GetPCRunning, 86  
GetPCSleep, 148  
GetPCSneaking, 86

- GetPCTraveling, 63
- GetPlayerControlsDisabled, 150
- GetPlayerFightingDisabled, 150
- GetPlayerJumpingDisabled, 150
- GetPlayerLookingDisabled, 150
- GetPlayerMagicDisabled, 150
- GetPlayerViewSwitch**, 150
- GetPos, 59
- GetRace, 90
- GetScale, 56
- GetSecondsPassed, 143
- GetSecundaPhase, 144
- GetSoundPlaying, 141
- GetSpell, 135
- GetSpellEffects, 135
- GetSpellReadied, 87
- GetSquareRoot, 157
- GetStandingActor, 64
- GetStandingPC, 64
- GetStat, 113
- GetTarget, 121
- GetVanityModeDisabled, 150
- GetWaterLevel, 157
- GetWeaponDrawn, 87
- GetWeaponType, 43
- GetWerewolfKills, 94
- GetWindSpeed, 147
- Global scripts, 23
- global* variables, 29
- Goodbye, 108
- GotoJail, 126

## H

- HasItemEquipped, 45
- HasSoulgem, 131
- Help, 216
- HitAttemptOnMe, 122
- HitOnMe, 122
- HurtCollidingActor, 65
- HurtStandingActor, 65

## I

- If, 31
- indentation, 26
- INFIX to POSTFIX**, 206
- IsWerewolf, 95

## J

- Journal, 110

## L

- LeftEval**, 206
- Local scripts, 23
- local* variables, 28
- Lock, 69
- Long, 27
- LoopGroup, 70
- LowerRank, 91

## M

- Mathematical calculations, 30
- maximum health, 114
- MenuMode, 88, 151

- MenuTest, 152
- MessageBox, 103
- minimumprofit, 88
- ModCurrentFatigue, 114
- ModCurrentHealth, 114
- ModCurrentMagicka, 114
- ModFactionReaction, 93
- ModHealth, 113
- ModPCFacRep, 93
- ModRegion, 146
- ModScale, 56
- ModStat, 113
- ModWaterLevel, 157
- Morrowind Enhanced, 161
- Morrowind Graphics Extender, 161
- Morrowind Script Extender, 161
- moto, 216
- Move, 46
- MoveOneToOne, 216
- MoveWorld, 47
- Multiplication, 30

## O

- Objects**, 15
- OnActivate, 66
- OnDeath, 123
- OnKnockout, 124
- OnMurder, 123
- OnPCAdd, 39
- OnPCDrop, 39
- OnPCEquip, 41
- OnPCHitMe, 120, 127
- OnPCRepair, 43
- OnPCSoulGemUse, 39, 132
- OnRepair**, 43
- operators, 30
- OR, 34

## P

- PayFine, 126
- PayFineThief, 126
- PC Clothing Modifier (dialogue), 111
- PC Sex (dialogue), 111
- PCClearExpelled, 92
- PCExpell, 92
- PCExpelled, 91
- PCForce1stPerson, 150
- PCForce3rdPerson, 150
- PCGet3rdPerson, 150
- PCHasGoldDiscount, 128
- PCJoinFaction, 91
- PCLowerRank, 91
- PCRaiseRank, 91
- PCSkipEquip, 42
- PlaceAtMe, 52
- PlaceAtPC, 51
- PlaceItem, 52
- PlaceItemCell, 52
- PlayBink, 155
- PlayGroup, 69
- PlaySound, 141
- PlaySound3D, 141
- PlaySound3DVP, 141
- PlaySoundVP, 141
- Position, 50
- PositionCell, 50

PT, 216  
Purge textures, 216

## R

RaiseRank, 91  
Random, 154  
Rank Requirement (dialogue), 111  
References persist, 165  
RemoveEffects, 136  
RemoveFromLevCreature, 156  
RemoveFromLevItem, 156  
RemoveItem, 36  
RemoveSoulgem, 131  
RemoveSpell, 133  
RepairedOnMe, 43  
Resurrect, 125  
Return, 153  
**RightEval**, 206  
Rotate, 54  
RotateWorld, 54

## S

SA, 218  
SameFaction, 91  
Say, 140  
SayDone, 140  
scripting window, 13  
ScriptRunning, 153  
SCT, 216  
Set ... to, 29  
SetAngle, 54  
SetAtStart, 51  
SetDelete, 72  
SetFactionReaction, 93  
SetJournalIndex, 110  
SetPCFacRep, 93  
SetPos, 48  
SetScale, 56  
SetStat, 113  
SetWaterLevel, 157  
SetWerewolfAcrobatics, 94  
SetWillpower, 113  
SG, 218  
Short, 27  
Show, 216  
Show Animation, 218  
Show target group, 218  
ShowGroup, 218  
ShowMap, 154  
ShowRestMenu, 148  
ShowScenograph, 218  
ShowVars, 216  
sine, 189  
SkipAnim, 71  
SSG, 218  
ST, 218  
StartCombat, 120  
StartScript, 153  
StayOutside, 89  
StopCellTest, 216  
StopCombat, 120  
StopScript, 153  
StopSound, 141  
StreamMusic, 140  
string variables, 27  
Subtraction, 30

sv, 216  
Syntax, 25

## T

T3D, 217  
TA, 217  
Talked to PC (dialogue), 111  
TB, 217  
TCB, 217  
TCG, 217  
TCL, 217  
TCS, 217  
TDS, 217  
TDT, 217  
TESAME, 163  
TestCells, 217  
TestInteriorCells, 217  
TestModels, 217  
Text defines, 105  
TFH, 217  
TFOW, 217  
TG, 217  
TGM, 217  
TKS, 217  
TM, 217  
TMS, 217  
ToggleAI, 217  
ToggleBorders, 217  
ToggleCollision, 217  
ToggleCollisionBoxes, 217  
ToggleCollisionGrid, 217  
ToggleCombatStats, 217  
ToggleDebugText, 217  
ToggleDialogueStats, 217  
ToggleFogOfWar, 217  
ToggleFullHelp, 217  
ToggleGodMode, 217  
ToggleGrid, 217  
ToggleKillStats, 217  
**ToggleLights**, 217  
ToggleLoadFade, 217  
ToggleMagicStats, 217  
ToggleMenus, 217  
TogglePathGrid, 218  
ToggleScriptOutput, 217  
ToggleScripts, 217  
ToggleSky, 217  
ToggleStats, 217  
**ToggleTextureString**, 217  
ToggleVanityMode, 218  
**ToggleWater**, 217  
ToggleWireframe, 218  
ToggleWorld, 217  
TPG, 218  
Tribunal Script Functions, 208  
Troubleshooting, 204  
TS, 217  
TSO, 217  
TST, 217  
TTS, 217  
TurnMoonRed, 94  
TurnMoonWhite, 94  
tutorial, 13  
TVM, 218  
TW, 217  
TWF, 218

## U

UndoWerewolf, 95  
Unlock, 69  
**UsedOnMe**, 46

## W

WakeUpPC, 148  
While, 33